

Solving Power System Analysis Problems Using Modern Software Approach

Mike Zhou, Ph.D.
Chief Architect
InterPSS Systems



Problem Statement

New advance in computer hardware and software technologies is enabling new ways to design power system analysis software to implement the existing and future models/algorithms in a flexible and extensible way to quickly evolve and adapt to future changes.

“Changing is the only constant” - Bill Gate



Setting the Stage

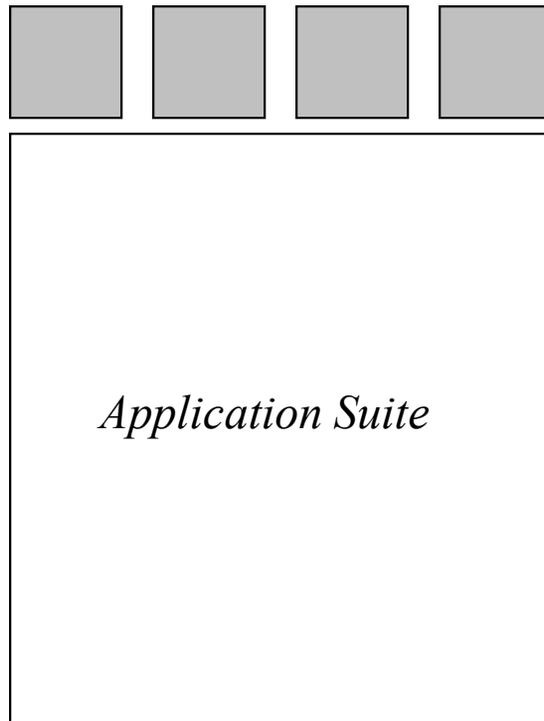
- **OFP models and algorithms**
- **Focus - Computer implementation of the model/algorithm**
 - Conceptual, architecture high-level
 - Common software concept and methodology
 - In-scope
 - Software architecture
 - Software flexibility and extensibility
 - Software maintainability
 - Out-scope
 - Applicability, accuracy and stability of the models/algorithms
 - Computation efficiency

Advances in Information Technology

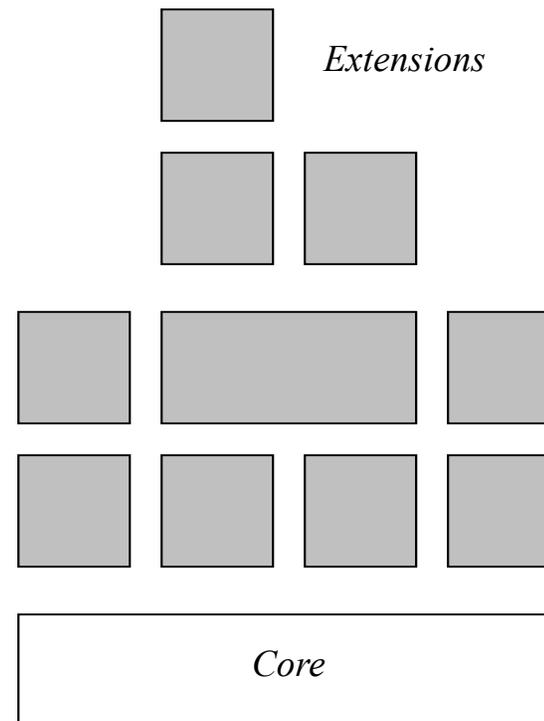
- **Network (10G pipe, 10M message/sec)**
- **Multi-processor, multi-core hardware architecture**
- **Virtualization technology**
 - Virtual machine technology (Java, C#)
 - Virtual operating system (VMWare, Amazon EC2 ...)
- **Software development**
 - Open-source – most advanced tools available for ordinary developers
 - Coding vs integration
 - MDA : Model Driven Application development
 - Code generation



Software Architecture – Then and Now



Traditional Approach



Component Approach

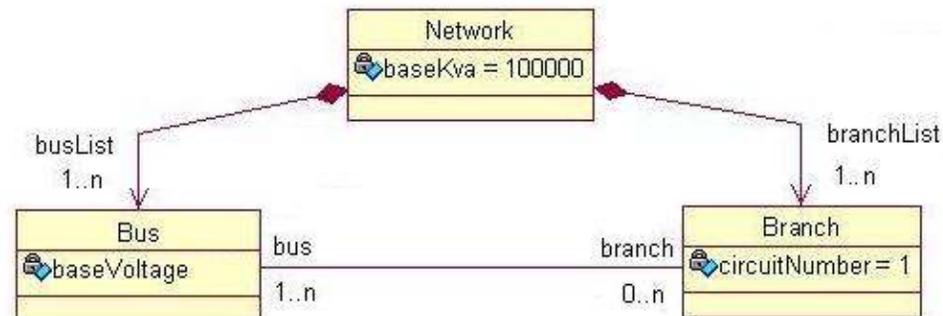
- **The Eclipse Project**

Model-based Approach

- Y-matrix – a conceptual model

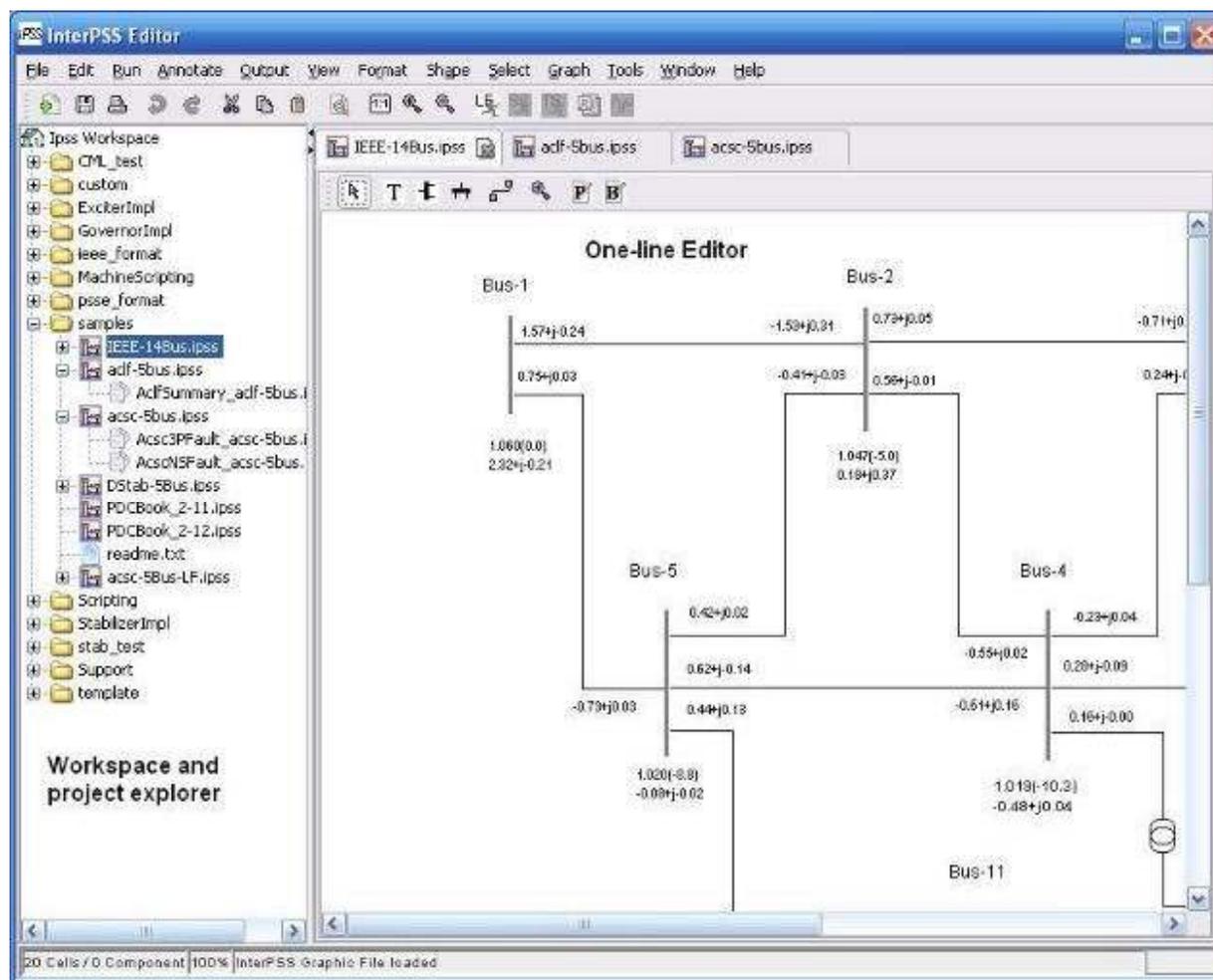
$$[V] = [Y] \times [I]$$

- Computer implementation model
 - Fortran style - One or two-dimensional array (50K-bus version)
 - Bus/Branch Model approach
 - Apply to sparse matrix
 - Code generation (MDA)

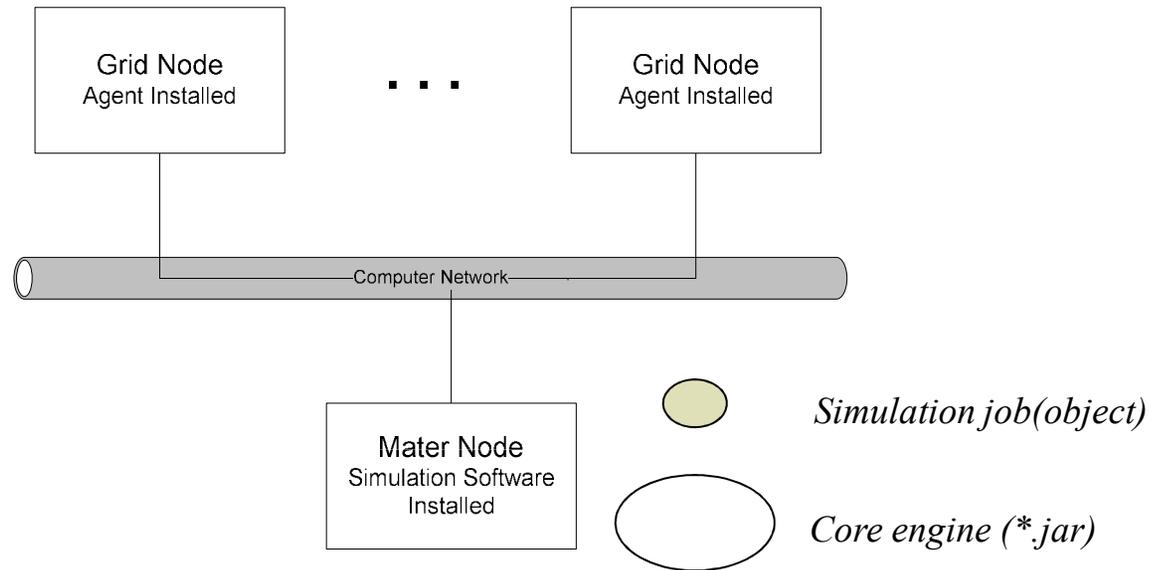


*E.Z. Zhou, 'Object-oriented Programming, C++ and Power System Simulation',
IEEE Transactions on Power Systems, Feb. 1996*

Desktop Application



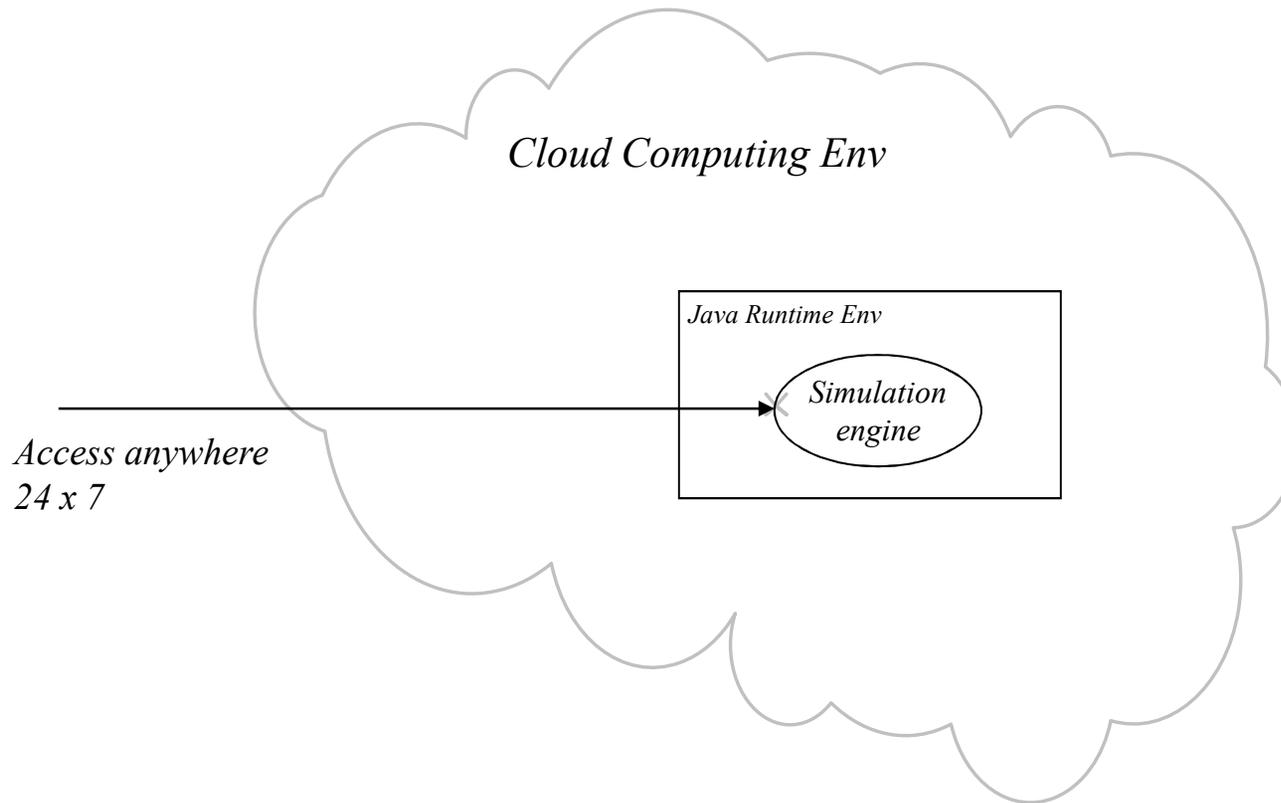
Grid Computing



- Simulation engine runs as a service
- Simulation algorithm implementation multi-thread safe
- Simulation job serialized/de-serialized as Xml document
- OS neutral language (Java) preferred



Cloud Computing



- **Cloud computing – in the process to become mature**
- **Power system – cloud of generation resources**



The InterPSS Project

▪ InterPSS

- Internet Technology-based Power System Simulation Software
- Apply modern software approach to solve power system analysis problems
- Java is chosen as its programming language

▪ Open architecture

- Components from other system could be integrated into InterPSS
- InterPSS components could be taken out and integrated into other systems



InterPSS Architecture

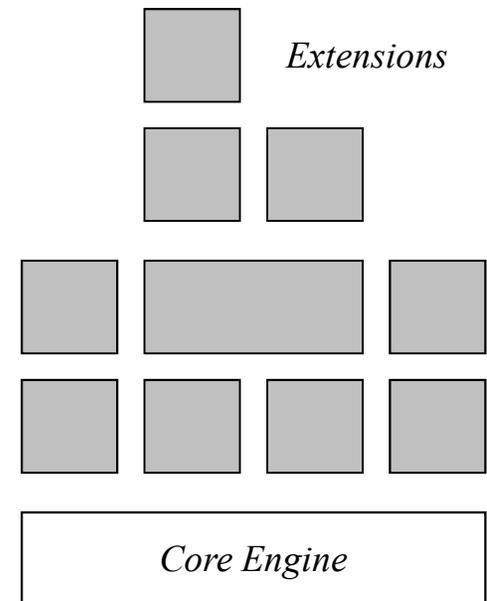
A good architecture is not there is nothing to add, rather nothing could be taken away

InterPSS Core Engine

Generic Power Network (Y-matrix) Solver

+ IoC Container (Spring Framework)

+ Default (replaceable) Algorithm Implementations



Component Approach

InterPSS Systems



“Simple yet Powerful”

- **Simple – a generic Y-matrix solver**
 - Reuse – any Y-matrix based algorithm
 - Model-based approach
 - 70% of InterPSS core engine code generated by computer
 - Multi-thread safe
- **Powerful – extension through IoC injection**
 - IoC (Inversion of Control)
- **Based on the same core simulation engine**
 - InterPSS desktop application
 - AC Loadflow, DC Loadflow/Sensitivity Analysis, Short Circuit, Transient Stability
 - InterPSS Grid Computing Edition
 - InterPSS Cloud Edition
 - InterPSS Interactive for Power Trading



InterPSS Project Info

- <http://community.interpss.org>
- **Started 2005**
- **Volunteers in US, Canada, China**
- **Open and free for any purpose**



Summary

- **Fundamental changes in software industry over the last 30 years**
 - Software runtime environment
 - Software development methodology
- **Smart grid needs support by “smart” power system analysis software**
 - Modern software approach
 - Valuable lessons
 - Through the InterPSS project.
 - Openly shared (<http://community.interpss.org>)

