



Using SansGUI[®] in Academia

**Raising Software Standards in Higher
Scientific & Engineering Education**

Copyright © 2002 ProtoDesign, Inc.

All rights reserved.



Overview

- **Academic Environment**
- **Software Requirements**
- **Software Tools**
- **The SansGUI Solution**
- **Development and Deployment**
- **Demonstration - SansGUI in Action**
- **Configurations and Support**

Academic Environment

■ Teaching

- Exposing students to emerging technologies
- Providing interactive learning environments

■ Research

- Concentrating on research subjects
- Employing leading edge software tools

■ Educational & Industrial Affiliation

- Publicizing and sharing research results

Current Problems

■ Primitive User Interfaces

- Command-line driven applications
- Text-base input and output files

■ No Interactive Schematic Editor

■ No Simple Mechanism to Integrate Existing Code with Modern GUI

■ No Standard for Project Specifications and Consolidation

Software Requirements ... 1 / 6

■ Graphical User Interface (GUI)

- Classifying model building blocks
- Configuring models with schematic diagrams
- Assisting data entry and validation
- Using familiar GUI operations
- Unifying coding and data accessing interfaces among instructors, students, and researchers
- Extending class work to future research
- Performance issues: simulation code and GUI

Software Requirements ... 2/6

■ Model Semantics

- Conveying the original intent of the model developer to the users
- Ensuring integrity of measuring units
- Encoding input data range semantics
- Generating comments from qualitative descriptions
- Incorporating context-sensitive documentation

Software Requirements ... 3/6

■ Data from Various Sources

- Integrating data filters, pre-processors, and post-processors
- Supporting column-major and row-major data storage orders of multi-dimensional arrays

■ Code from Various Sources

- Incorporating existing code and libraries
- Integrating with other software packages and environments

Software Requirements ... 4/6

■ Model Evolution & Version Control

- Cumulating models and simulation results from years of research and development
- Handling model specification changes
- Synchronizing simulation code and user data
- Sharing or merging software components, code, and data among projects
- Maintaining backward and, even better, forward compatibility

Software Requirements ... 5/6

■ Development Environment

- Supporting main-stream programming languages (C/C++, Fortran, Java[®], etc.)
- Integrating with main-stream software packages (MATLAB[®], MS Office[®], etc.)
- Providing verbose debugging facilities
- Concentrating on underlying logic and mathematics, not details of syntax or usage
- Supporting legacy code integration

Software Requirements ... 6/6

■ Web-Based Publishing

- **Constraints on web-based applications**
 - ◆ Form based data entry
 - ◆ Limited interactivity and data validation
 - ◆ Project management issues
- **Publication quality of diagrams, charts, and tables**
- **Consolidation of software documents**
- **On-line and off-line application versions**

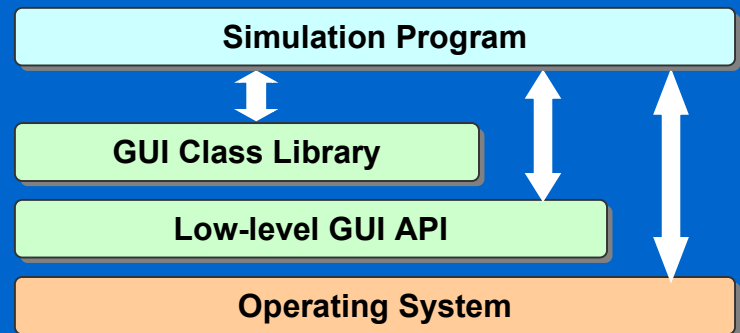
Academic Software Tools

- Office and Presentation Tools
- Internet Browsing / Authoring Tools
- Database & Data Access Tools
- Mathematical Packages & Libraries
- 3D Geometry & Visualization Tools
- Development Environments
- Domain Specific / Custom Tools

GUI Development ... 1 / 3

■ Low-Level API

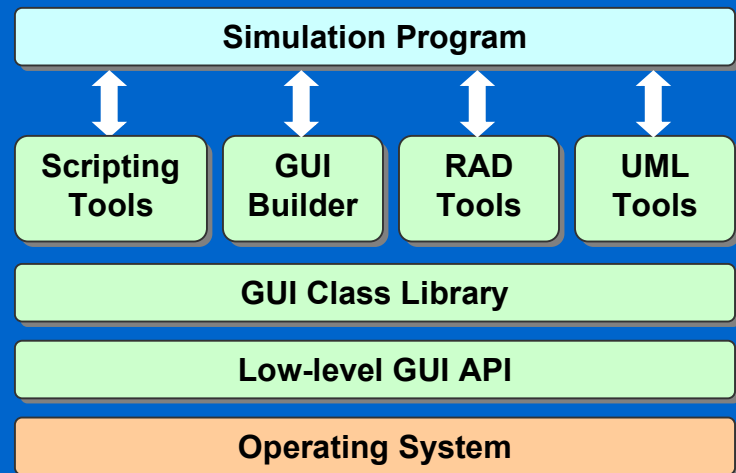
- **Microsoft Windows**
 - ◆ Win32, MFC, ...
- **Unix**
 - ◆ Xlib / Xt / Motif, Qt, ...
- **Java platform**
 - ◆ Swing, SWT, ...



GUI Development ... 2/3

■ Mid-Level API

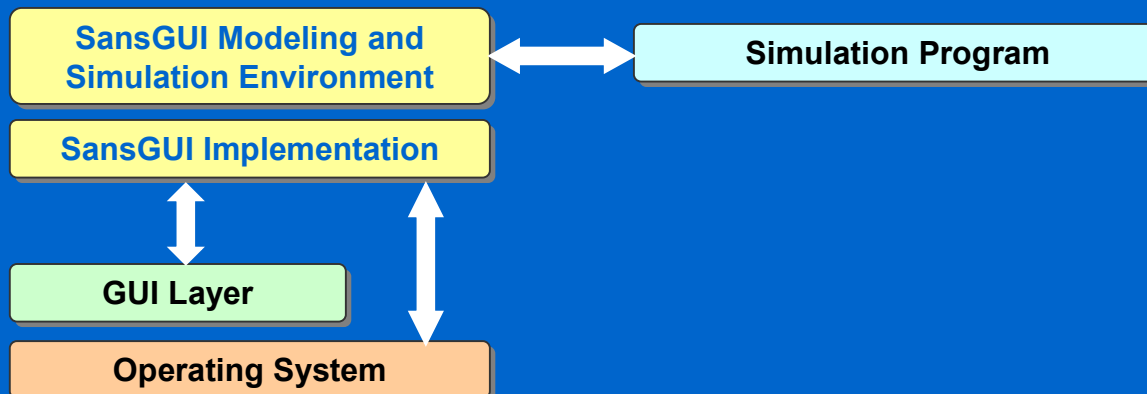
- Scripting tools
 - ◆ Tcl / Tk, Python, ...
- GUI builders with layout managers
- GUI components
- RAD tools
- UML GUI generation
 - ◆ MVC, MVP, ...



GUI Development ... 3/3

■ High-Level Environment

- Separation of GUI and simulator modules
- Simple communication protocol
- No GUI layout nor coding in development





The SansGUI[®] Solution



What is SansGUI ...

■ A Software Framework

- For developing and deploying simulation programs *without* tedious *Graphical User Interface* programming

■ An Interactive Environment

- For building models, entering parameters, running simulations, monitoring progress, and viewing results

Who are the users ...

Anyone who is involved in developing and using computer simulation in scientific and engineering fields.

■ Simulation Developer

- Professors / Instructors / Research Associates
Research Scientists / Research Engineers

■ Simulation User

- Students / Data Analysts / Design Engineers

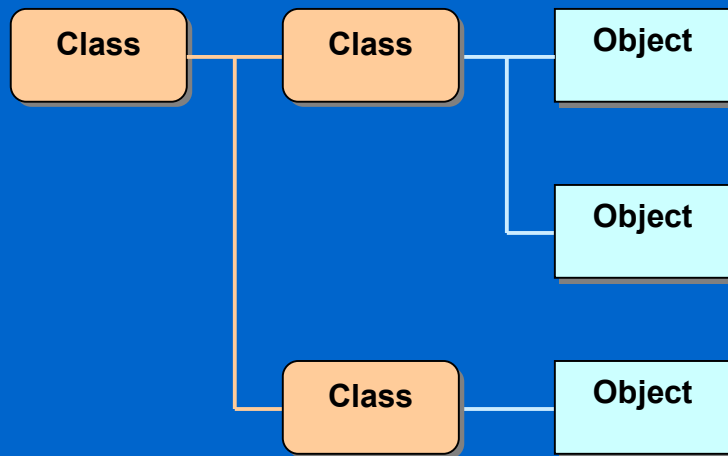
SansGUI Abstraction

■ SansGUI Object System

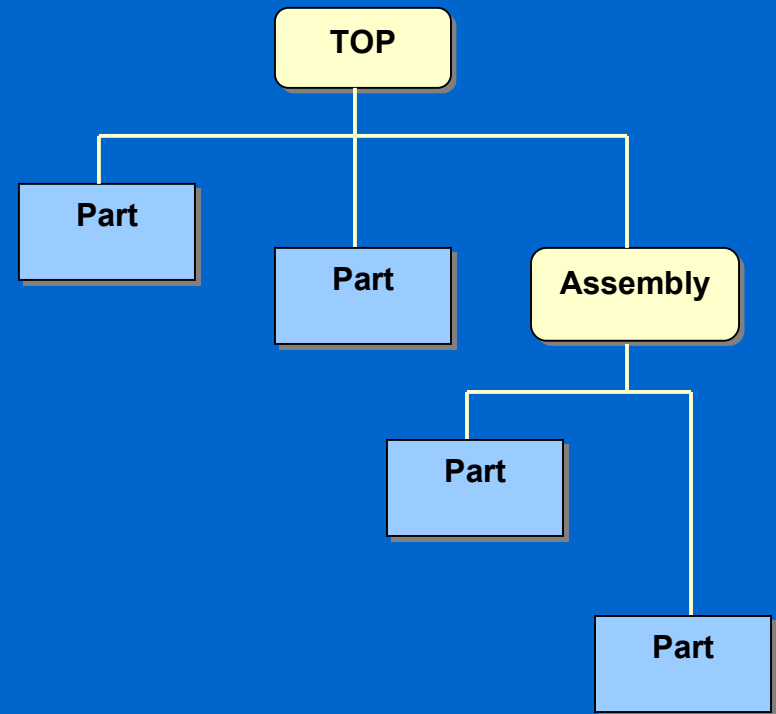
- **Hierarchical class schema definitions**
- **Attribute and behavior inheritance**
- **Component and connection objects**
 - ◆ Ports and connectivity definitions
- **Reference objects**
 - ◆ Collections, Matrices, and Tables
- **Simulation control objects**
- **User-extensible unit objects**

Project Models

■ Class Hierarchy



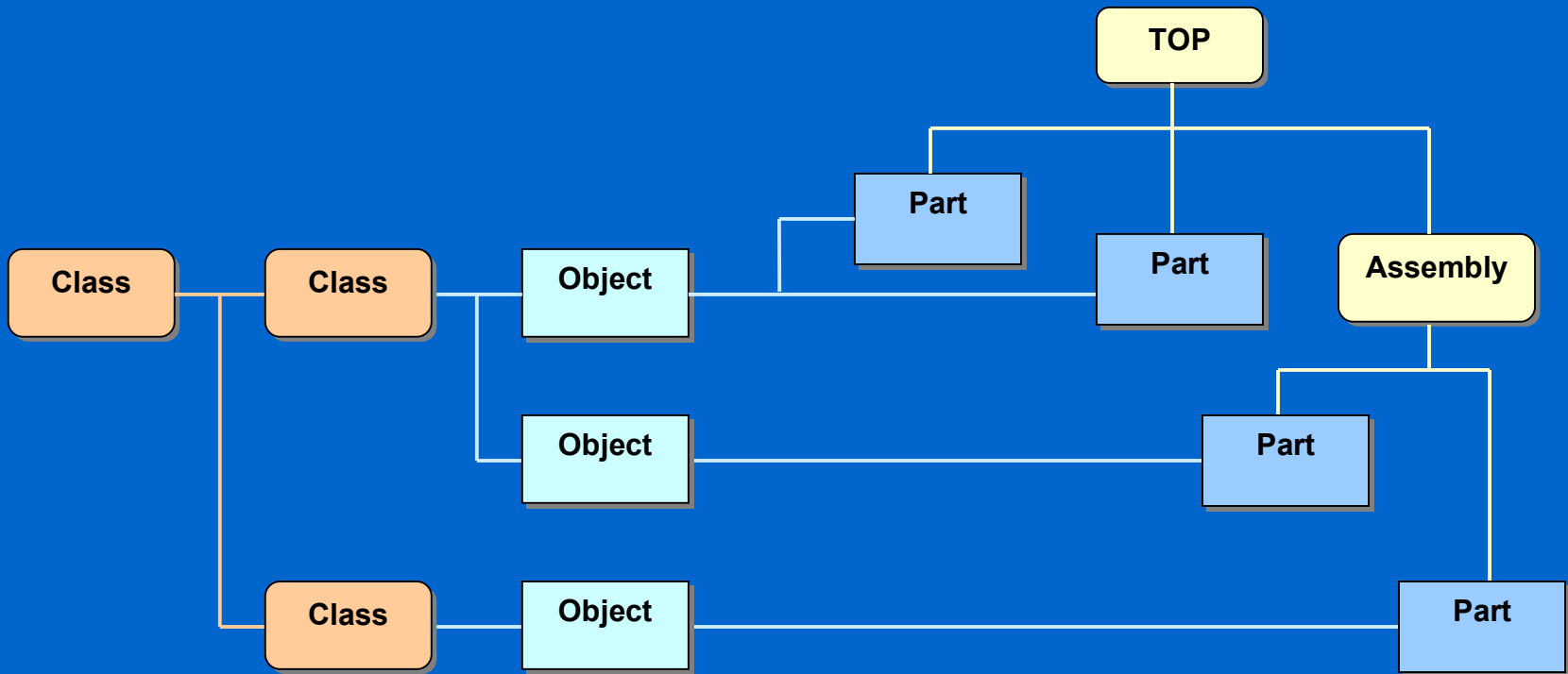
■ Assembly Hierarchy



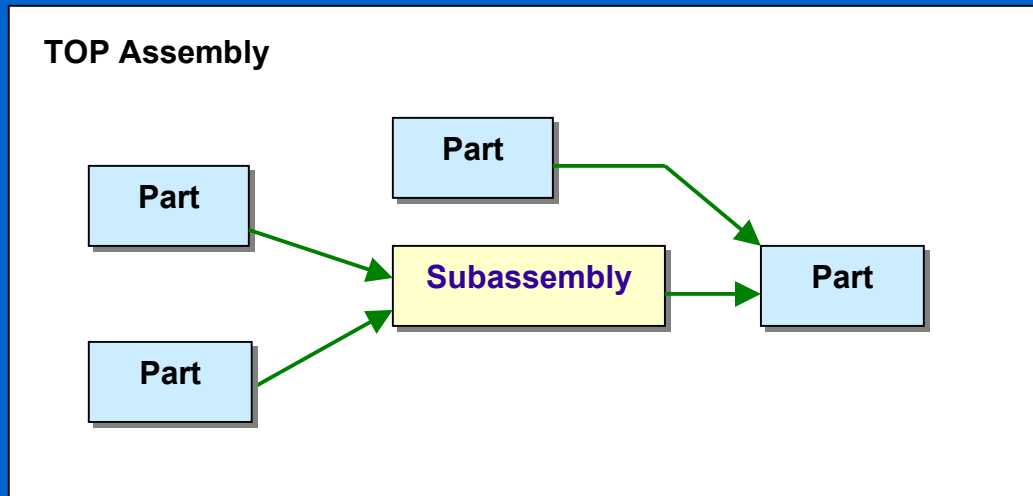
Project Models

■ Class Hierarchy

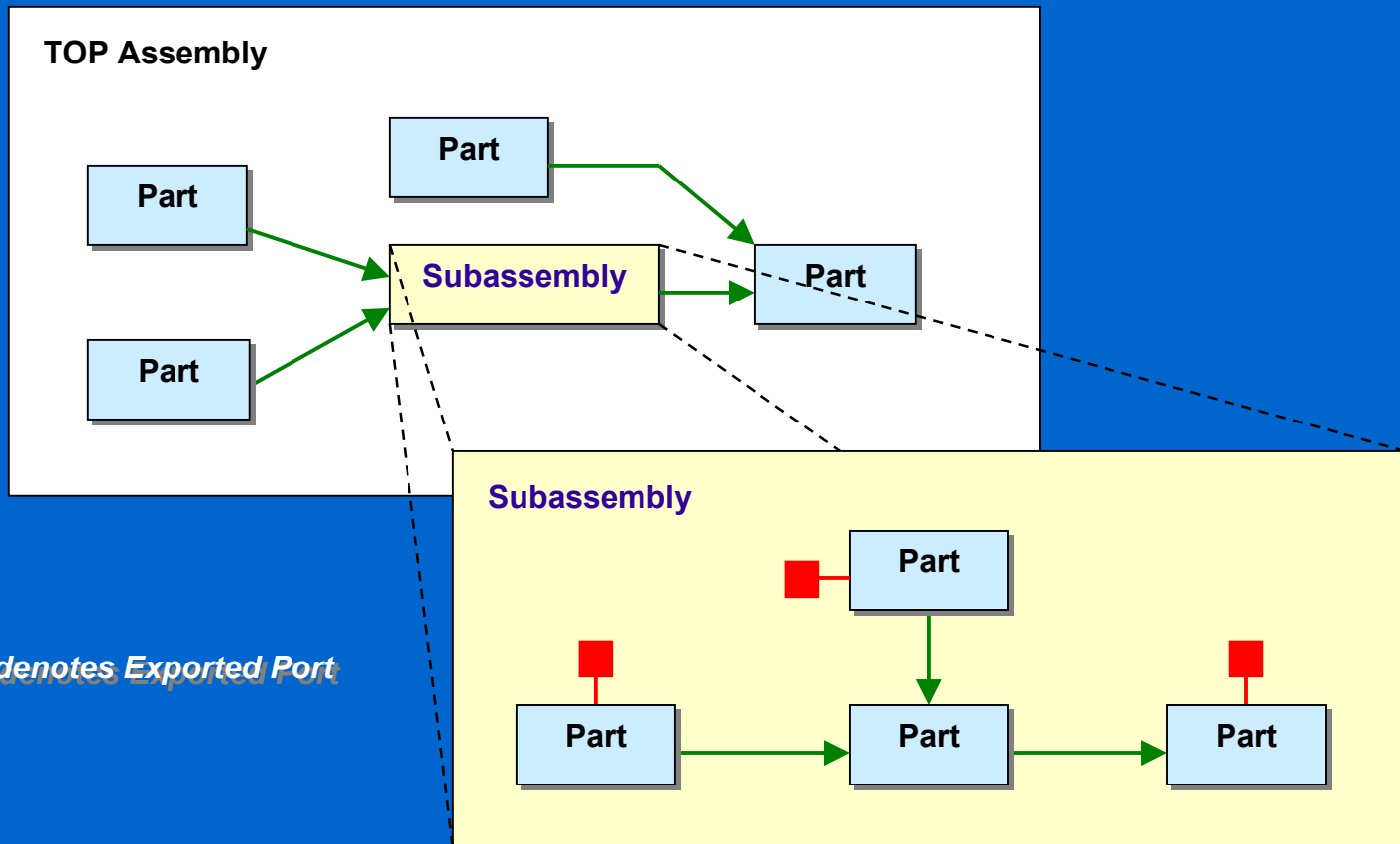
■ Assembly Hierarchy



Model Configurations



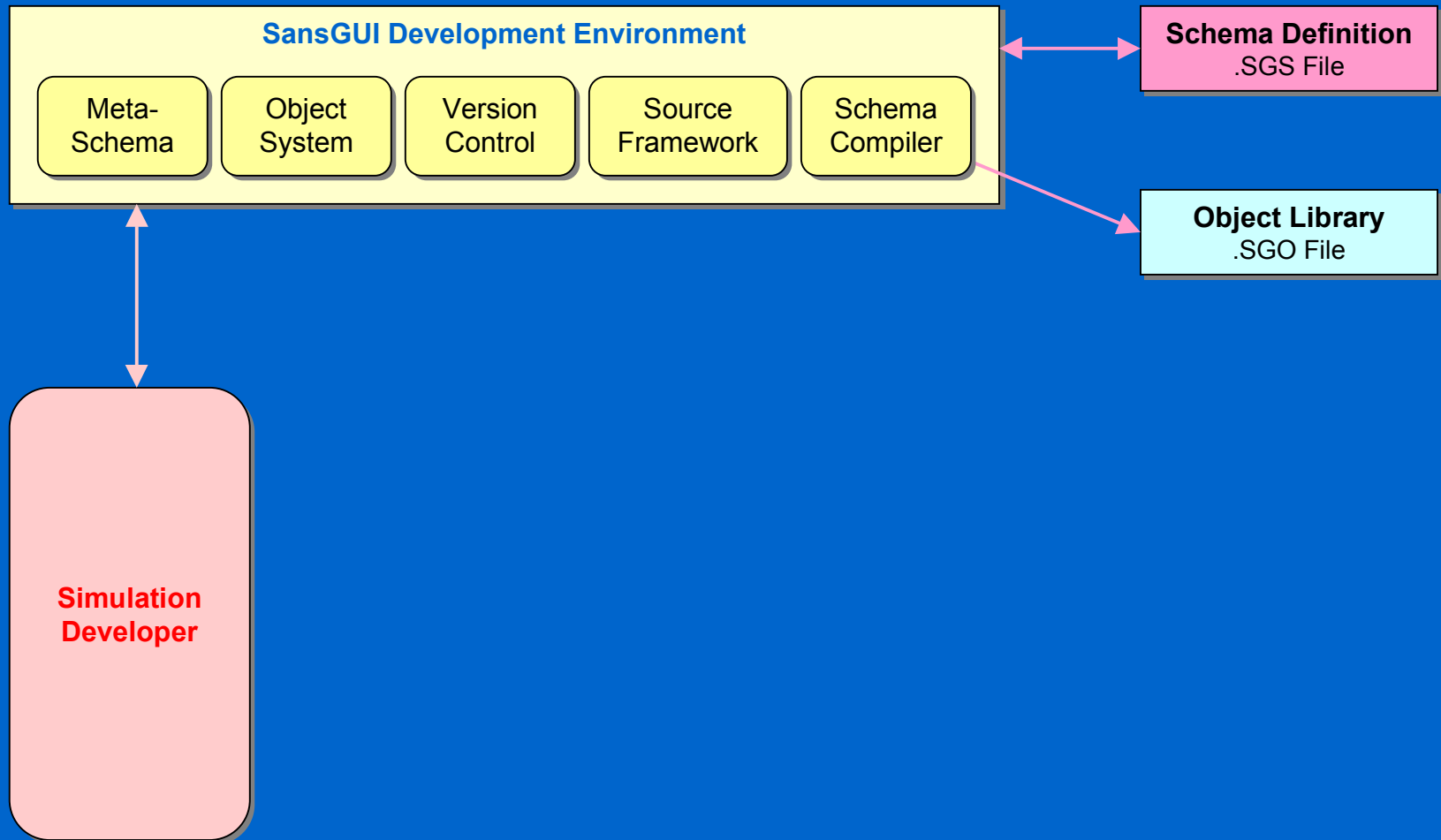
Model Configurations



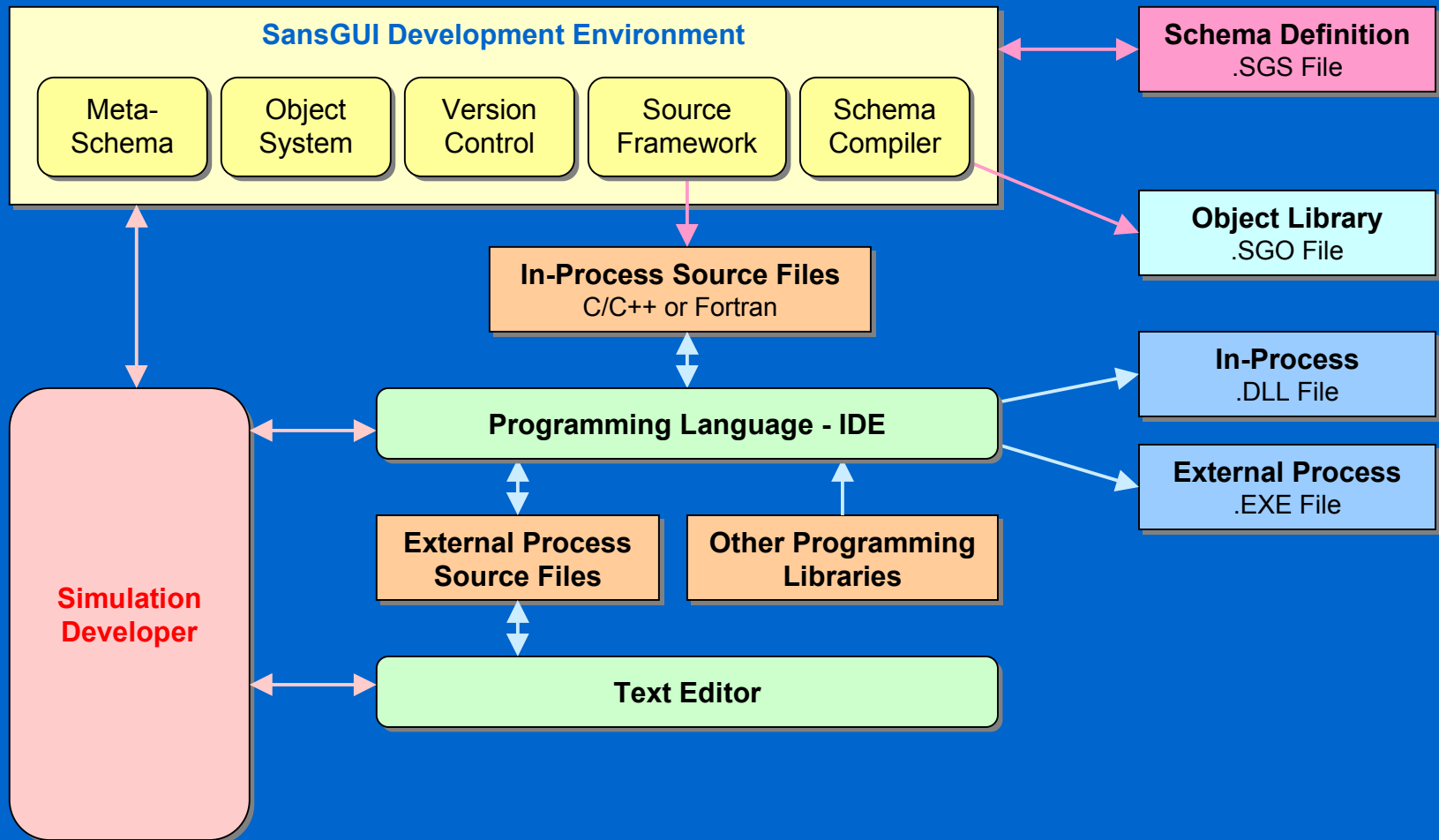
SansGUI Architecture

- **SansGUI Development Environment**
 - Define model building blocks in classes
 - Specify attributes involved in simulation model
 - Implement simulator logic and mathematics
- **SansGUI Run-Time Environment**
 - Configure model and prepare input data
 - Run simulation and monitor progress
 - Analyze simulation results to refine design

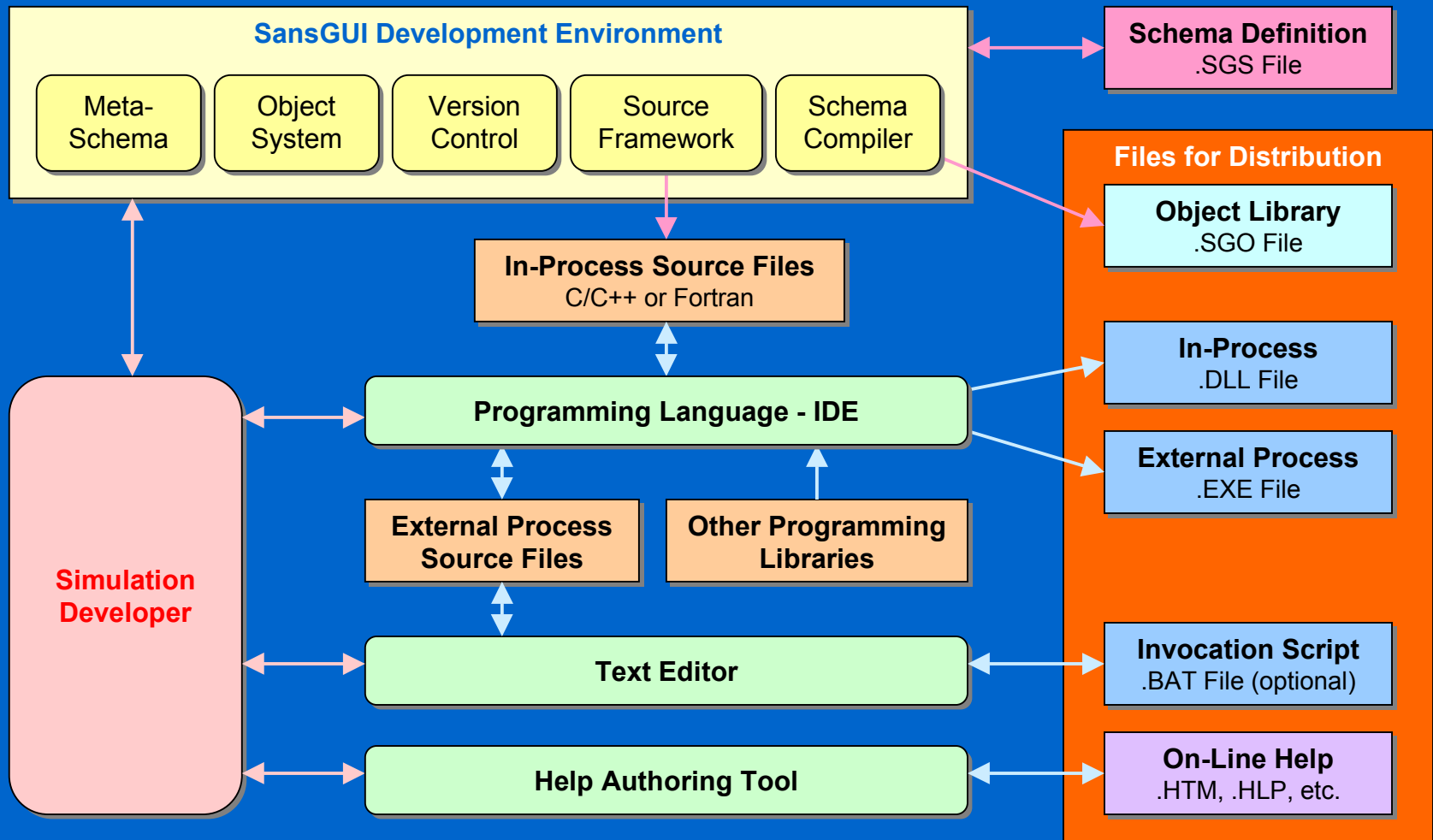
Development Environment



Development Environment



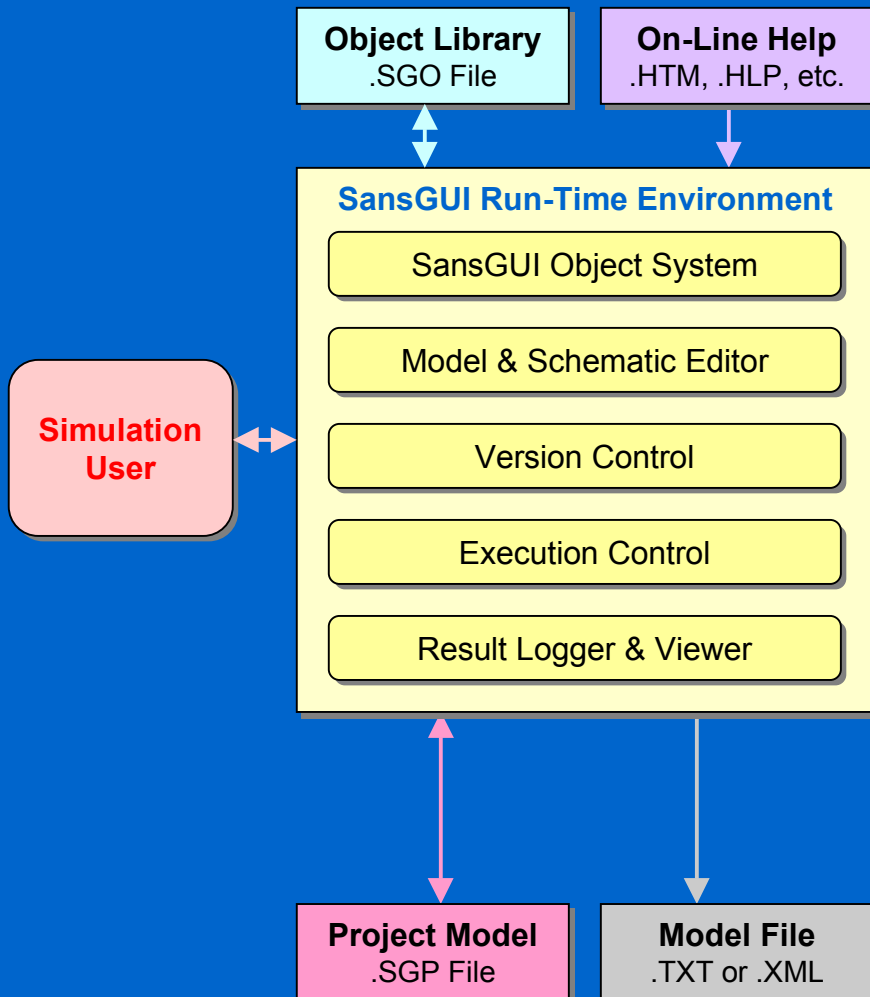
Development Environment



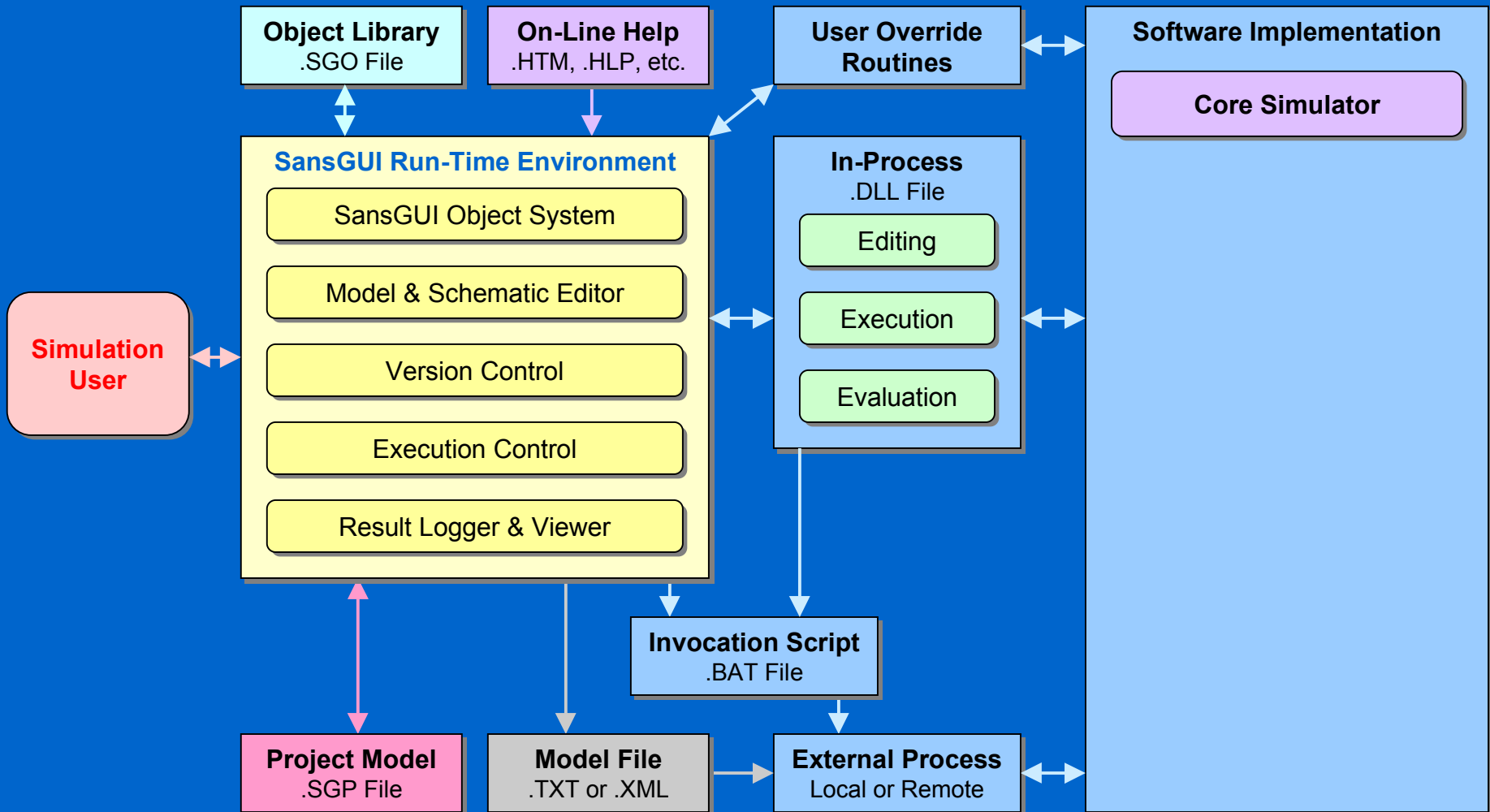
Development Features

- **Simulation Program Definition**
- **Class Schema Creation and Maintenance**
- **Class Persistence and Evolution**
- **Unit Object Creation & Maintenance**
- **SansGUI Source Code Framework**
- **Interactive Tracing and Debugging**

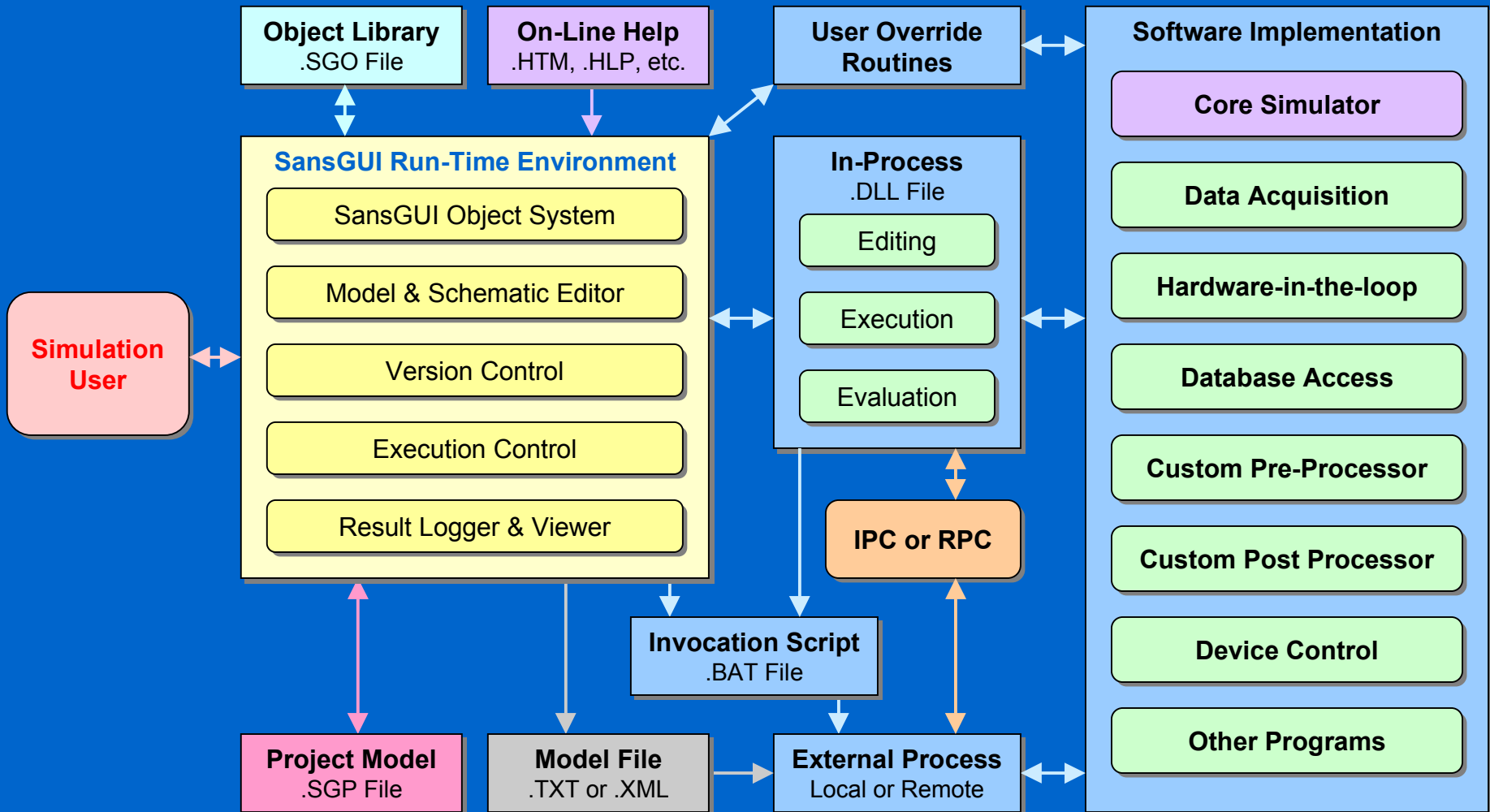
Run-Time Environment



Run-Time Environment



Run-Time Environment



Run-Time Features

- **Object Creation and Maintenance**
- **Data Entry Assistance and Validation**
- **Interactive Model Building Features**
- **Model Data Persistence and Evolution**
- **Parameter Resolution**
- **Simulation Control**
- **Dynamic Charting**

Deployment

■ Object Library

- Default or customized with convenient objects

■ Executable Code in .DLL or .EXE

■ Invocation Script (Optional)

■ On-Line Documentation

■ Example Project Models

■ Examples on ProtoDesign Web Site



SansGUI[®] in Action

Supported Languages

- **Microsoft Visual C++ 6.0 (SP3) +**
 - Earlier versions may still work
- **Compaq Visual Fortran 6.1a +**
- **DLL Project Creation**
- **Interactive Code Debugging**
- **XML Model Data Support**
- **Class Schema Version Control**

Other Language Support / PC

■ In-Process Simulation

- Can create Win32 DLLs
- Can be called from Microsoft Visual C++
 - ◆ Function names (length and case sensitivity)
 - ◆ Compatible function argument list
- Can access C data structures with pointers

■ External Process Simulation

- Can create stand-alone programs
 - ◆ Batch command / Shell script invocation

Licensing SansGUI

■ **SGpro - Professional Edition**

- Full featured Development Environment

■ **SGstu - Student Edition**

- Development Environment with 10 classes and 20 attributes per class limitation

■ **SGrun - Run-time Edition**

- No Schema Definition access

■ **SGdemo - Run-time Demo, No Save**

Configurations for Teaching

■ **Classes Requiring Programming**

- **Instructor: Professional Edition**
- **Students create new classes: Student Edition**
- **Students use predefined classes: Run-Time**

■ **Classes without Programming**

- **Instructor: Professional Edition**
- **Students: Run-Time Edition**

■ **Playback Only: Free Demo Edition**

Configurations for Research

■ Group Research

- Laboratory computers: Professional Edition
- Group leaders: Professional Edition
- Research users: Run-Time Edition

■ Individual Research

- Professional Edition

■ Colleagues and Interested Parties

- Run-Time Edition or free Demo Edition

Educational Grants*

■ **Course Adoptions**

- **Verifiable course information**
 - ◆ School, Department, Professor(s)
 - ◆ Course Number, Title, Description, Term, Web Link
- **One Professional Edition to instructor**
- **Free Evaluation Licenses to students**

■ **Special Projects**

- **Specially arranged with ProtoDesign, Inc.**

*Grant terms subject to change without notice.

Thank You !

■ Visit Our Web Site

- <http://protodesign-inc.com>
- <http://sansgui.com>

■ E-Mail

- Information: info@protodesign-inc.com
- Sales: sales@protodesign-inc.com
- Support: support@protodesign-inc.com
- Beta Testing: beta@protodesign-inc.com