LIBPF: A LIBRARY FOR PROCESS FLOWSHEETING IN C++

Barcelona, October 4th 2006

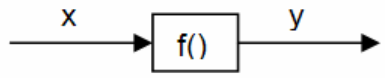
Contents

What is Process Flowsheeting ?

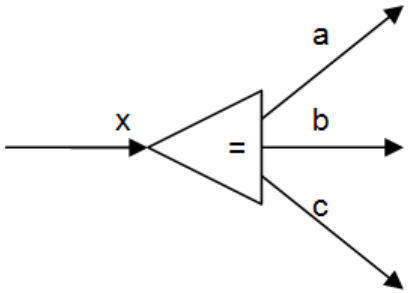
- Concentrated parameters, deterministic modelling of a continuous process based on a directed graph
  - edges are {material, signal, energy} streams
  - vertexes are transformations on streams
- See Westerberg et al. 1974

Any equation (set) can be seen as a directed graph

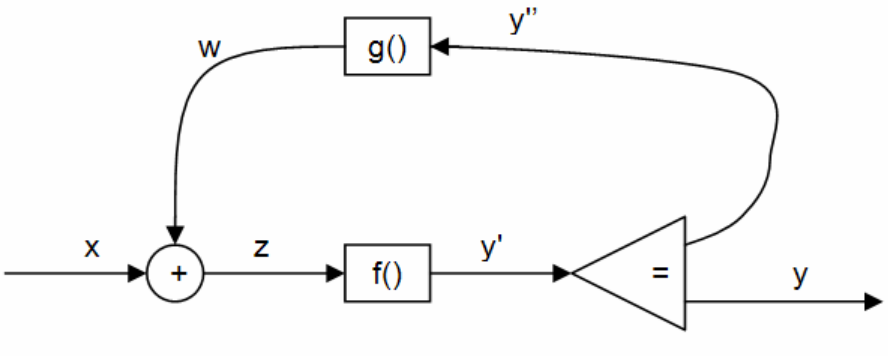
$$y = f(x)$$



$$\begin{cases} a = x \\ b = x \\ c = x \end{cases}$$



$$\begin{cases} z = x + w \\ y' = f(z) \\ y = y' \\ y'' = y' \\ w = g(y'') \end{cases}$$



Process flowsheet example

domain:

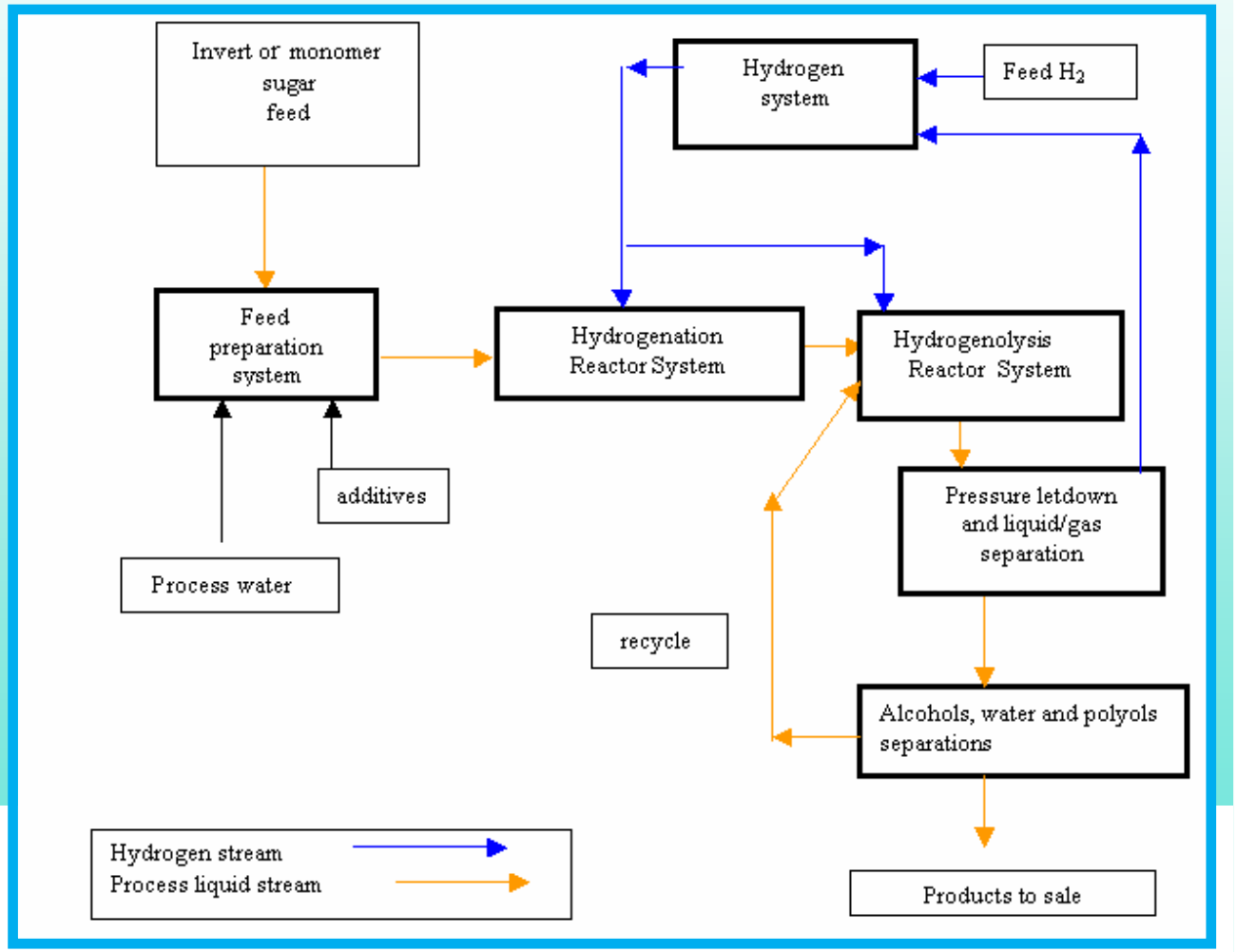
chemical

engineering

process:

polyols

from biomass



Other names for Process Flowsheeting...

- Stock-flow diagrams in econometric models
- “network structure” in LCA (Life Cycle Assessment)
- Heating Ventilation and Air Conditioning (HVAC) modelling

State of Modeling of Continuous Processes in the Industry

- Required skills:
  - Modeling
  - Software Engineering
  - Process knowledge

Challenges

Large unexploited potential

Challenges

- Single or small-series realizations of processes
- Complexity
- Safety and reliability
- Many different models used by different people in different phases of the project

Many different models 1/2

Feasibility study

Conceptual process design

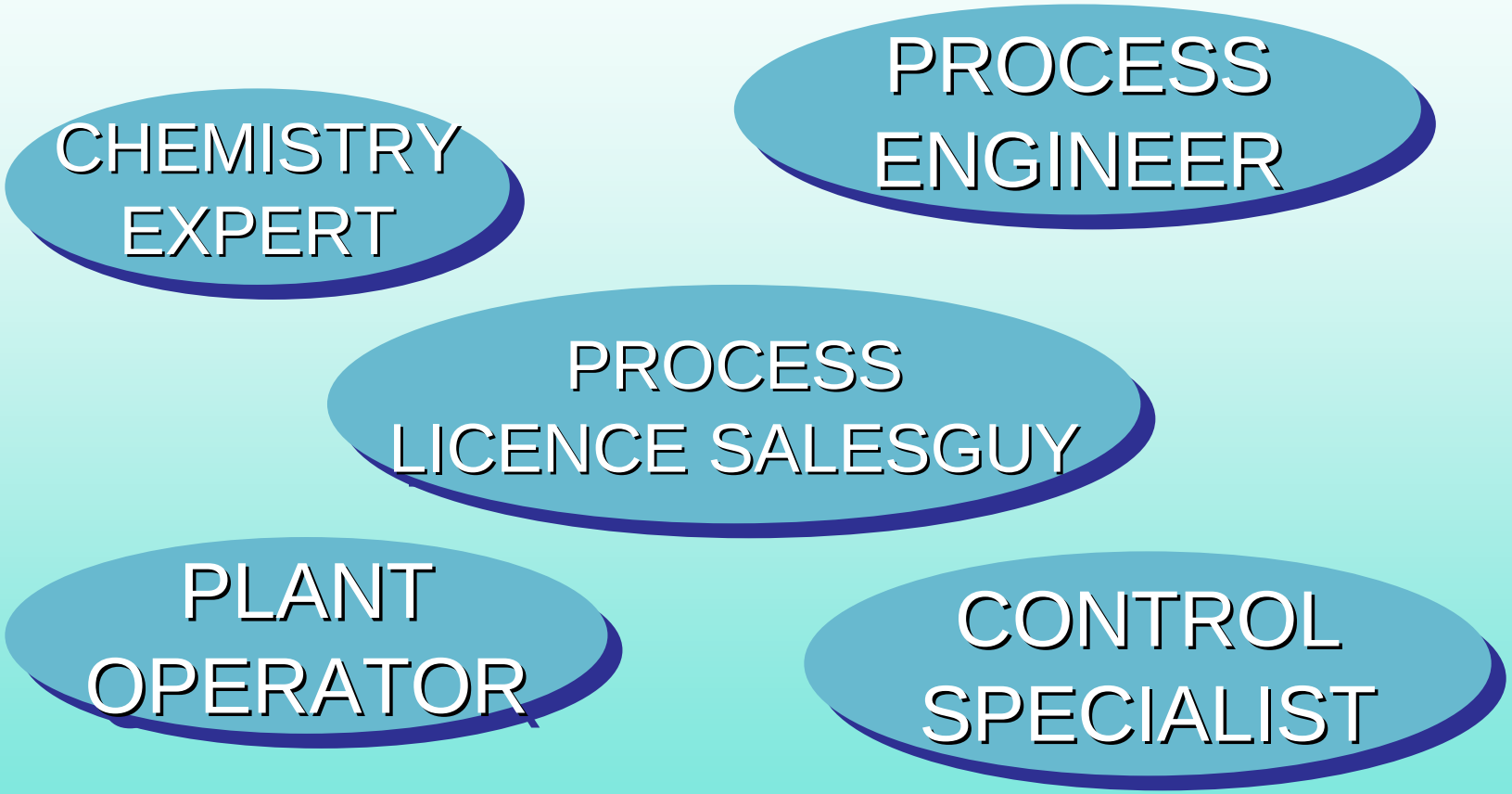
Basic of pilot plant

Pilot plant data reconciliation processing and interpretation

Basic engineering of production unit

Operations

Many different models 2/2



Requirements for industrial modelling solutions

- Customization
- Integration
- Reliability
- Maintainability

Req 1: Customization

- Each project will be different
- In terms of solution provider, development tool has to be flexible
- But limit project cost **otherwise it will be impossible to enter the market**

Req 2: Integration

- Need for interfaces: OPC, ODBC
- Need to support different hardware:
  - Workstation
  - DCS
  - Industrial PC
- Need to support different operating systems

Req 3: Reliability

- Provide correct results if solution exists
- Provide and log errors (communication, data consistency, computation)
- Never crash
- No memory leaks (for long execution times)

Req 4: Maintainability

- Short term: low cost to fix bugs
- Long term (25 years): can upgrade, update and recompile
- Own or can freely access source code
- Own or can freely access development tools

Tools & Options

Commercial process simulators: gProms, ACM, HySys, AspenPlus, PROII, ChemCAD ...

Mathematical toolboxes: Matlab, Mathematica...

Spreadsheets (!)

Programming languages: C++, FORTRAN, Java, Phyton, ObjectPascal ...

Commercial process simulators

pros	cons
■ Short development time ■ Libraries of models available ■ Can compile	■ Dependence on tool provider ■ Can do no real research ■ Small user community = big bugs

Mathematical toolboxes

pros	cons
■ Reliable and easy to use ■ Libraries of models and control algorithms available ■ Can compile via C converters	■ Dependence on tool provider ■ Interpreted ■ Objects and data structures not designed for chemical engineering

Programming languages

pros	cons
■ Language is vendor-independent, international standard = Portability ■ Can be faster	■ Tough and risky ■ Steep learning curve ■ Maintainability ?

LIBPF

- Description and Scope
- Capabilities and Applications
- Design
- License

What is LIBPF ?

- **LIBPF** = C++ **LIB**rary for **P**rocess **F**lowsheeting
- A collection of **objects and methods** to streamline the modelling activity
- Resolution of NLAE (Non-Linear Algebraic Equation) and DAE (Differential Algebraic Equations)
- Version 0.6, 30000 Lines Of Code (LOC)

Scope

- General purpose
- Simple models
- First principle (mass and energy balances, equilibria, rating relations)
- Concentrated parameters
- Modelling of whole processes (flowsheet)

Levels of modelling

- semiempirical, local: rule of thumb, soft sensor
- **LIBPF**
- first principle, system: concentrated parameters, entire process
- first principle, local: CFD, detailed design of single unit

Capabilities 1/3

- Components:
  - fluids
  - biotech (protein, lipid, carbohydrate, ash)
- Properties:
  - ideal vapor-liquid (dilute systems)
  - SRK equation of state

Capabilities 2/3

- Unit operations:
  - mixer, 2 or more inlets
  - flow splitter (tee), 2 or more outlets
  - spawn (duplicates the inlet)
  - fixed-yield separator, 2 or 3 outlet streams
  - vapour-liquid flash
  - isentropic compressor/expander
  - reactive **multi-stream** heat exchanger
  - fuel cell
  - countercurrent non-reactive adiabatic HTU/NTU column
  - **multistage units obtained combining any of above**

Capabilities 3/3

- Flowsheet resolution:
  - Supports feedback specifications
  - Sequential (direct substitution) or
  - **Simultaneous**

Applications

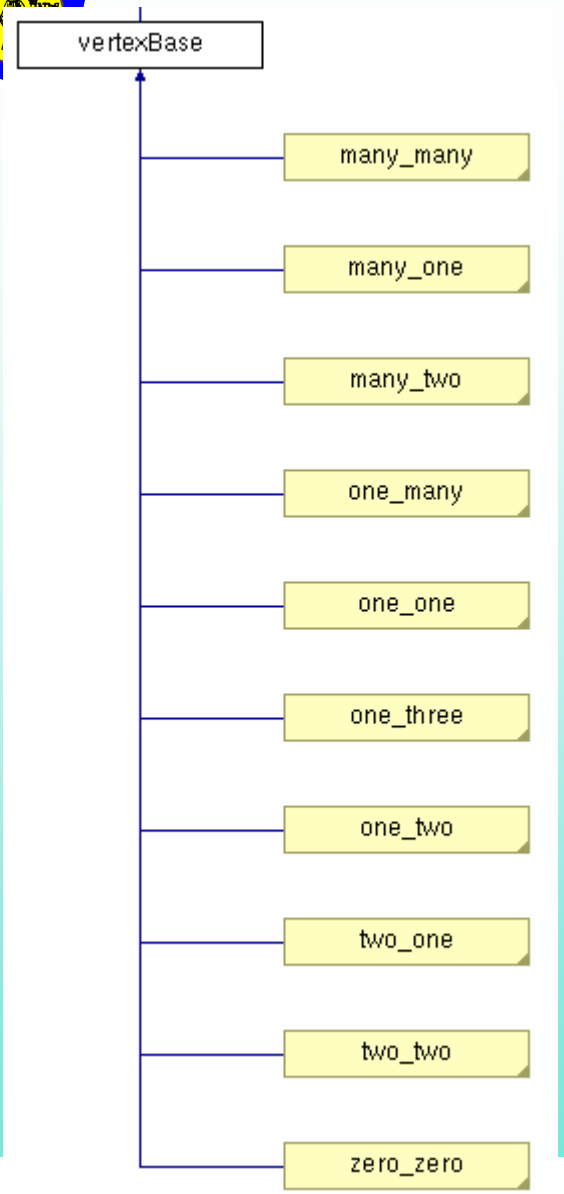
- Fuel cell system modelling
- Absorption/stripping
- Low pressure gas cleaning / processing
- Biotech processes

LIBPF design

- Flowsheeting in C++
- Portability
- Persistency to external database
- Small footprint salculation kernel
- Analytical derivatives
- Dimensional check of equations

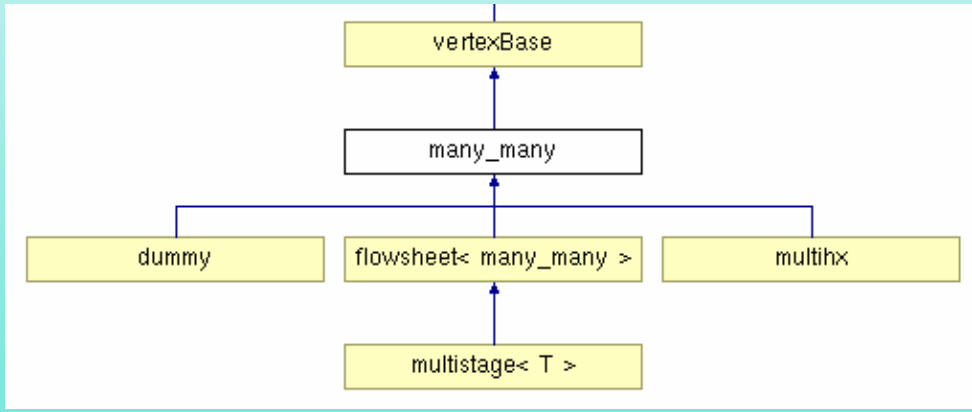
Flowsheeting in C++

- A flowsheet is a parameterized graph
  - Edges = Streams
  - Vertexes = Blocks
- Can use graph algorithms to analyze connectivity, find solution path



Vertex taxonomy

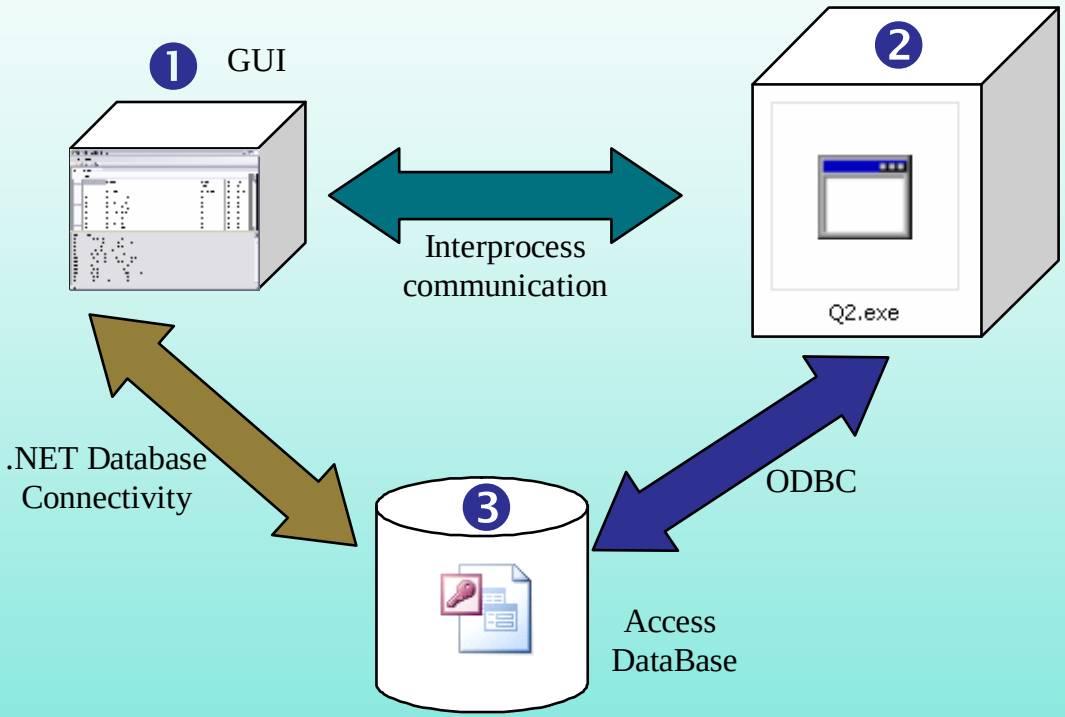
- Vertex models inherit from connectivity capability
- flowsheets can be vertexes in turn



Portability

- International Standard *"Programming Language C++" ISO/IEC 14882:1998*
- **Mac OSX** 10.2.8; GNU gcc 4.0.1
- **Windows** XP Professional SP2; GNU gcc 4.0.1, Microsoft Visual C++ 2005
- Debian **Linux** 3.1; Intel C Compiler 9.1, GNU gcc 4.0.2

Persistency to database



1. User Interface
2. Calculation kernel: console application from C++ source
3. Relational Database

Calculation kernel

- Small footprint: 1 ~ 4 MB
- Standalone, no weird dependency
- Can be installed on industrial PC, i.e. Windows XP Embedded

Analytical derivatives

- In LIBPF derivatives are not obtained with numerical perturbation (finite differences)
- Derivatives are analytical, obtained without source transformation via operator overloading
- Sparse and dense derivatives supported

Dimensional check of equations

- We want reliable engineering computations
- Options for dimensional consistency check:
  - compile-time using template metaprogramming, very slow compile
  - run-time, slows execution but can be turned off for production executable

LIBPF

Dimensional check of equations

```
Qdouble v;                              // the dimension is not known yet

Qdouble P(101325, "Pa");                // pressure
Qdouble T(298.15, "K");                 // temperature
Qdouble R(8314.4, "J/(kmol*K)");        // gas constant

v = R * T / P;                          // now v becomes a molar volume

diagnostic(0, "Molar volume = " << v);

v = Qdouble(0.0, "kg/m3");              // error !
```

```
main *** Molar volume = 24.4652 kmol^-1 m^3
UOM error in function: operator=
terminate called after throwing an instance of
    'errorUOM'
```

License

- Open source approach unsuitable
- Free academic license
- Flexible commercial licencing options

Open source approach unsuitable

- Open source does not stimulate innovation
- User community is too small
- “Hard” open source is not compatible with industry confidentality requirements
- Current open source projects struggling (ASCEND, SIM42, OpenSim)

Free academic license

- Researchers can get compiled form (DLL/LIB) of the library complete with headers and examples
- Allowed teaching and research, but commercial uses not allowed
- Objectives:
  - Increase impact
  - Test on the field
  - Create a community

Conclusions

LIBPF can do Process Flowsheeting in C++, no need for extra tool

LIBPF can help manage the entire life cycle of a modeling solution

Flexible licencing, inclusive free academic license

Visit www.libpf.com !