



This document is part of LIBPF

Paolo Greppi - 3iP © Copyright 2007 - 2008

All rights reserved; do not distribute without permission.

# **LIBPF Roadmap**

## **Summary**

Introduction.....	2
Critical system requirements and targets.....	2
Milestone plans.....	2
Version 0.9.....	3
Version 1.0.....	4



## Introduction

This roadmap describes the planned capabilities of **LIBPF**, the C++ **LIB**rary for **P**rocess **F**lowsheeting. The current version of LIBPF at the time of writing is 0.8.

### *Critical system requirements and targets*

LIBPF is a modelling tool for industrial, continuous processes. Its purpose is to make it possible to rapidly prototype and deploy small-footprint custom applications implementing computations for training, process engineer support, on-line process diagnostic, and data reconciliation. The necessary models (physical properties and unit operations) and tools (solvers, input/output) are gathered in the form of a C++ library; by linking to the LIBPF library it is possible to create lightweight, special-purpose applications devoted to compute a specific process or process family.

The objective of the continuing LIBPF development efforts is to enlarge the subset of real-life processes it can be applied to, increase its efficiency and improve the user friendliness.

Along this path the following guidelines will be enforced:

1. Target easy-to-implement, high-return features first;
2. Avoid feature bloat and contain code size;
3. Retain portability;
4. Avoid dependency on monolithic libraries and technologies;
5. Maximize model reuse through object-oriented and generic programming.

## Milestone plans

The planned revision schedule is based on a 6-months revision cycle. There are six **major areas of development**:

1. User interface
2. Modelling
3. Physical properties
4. Resolution
5. Interfaces
6. Systems



For each area of development the plan is to implement the indicated points.

## **Version 0.9**

Estimated deadline: 2009/02/28

Key deliverables:

- User interface:
  1. Port to more languages: es, ru, ar, he, ko, jp, zh;
  2. Display key data on streams and units in PFD tab on hovering;
  3. Filter database views with regexp;
- Modelling:
  1. Merge multihx and hx models;
  2. Kinetic-controlled reactors;
- Physical properties:
  1. Liquid-phase reactive Flash (i.e. formaldehyde-methanol-water system or aqueous ionic species);
  2. Three-phase Flash (VLLE);
  3. Handling of metastable roots in equation of state - Bug #56;
- Resolution:
  1. Improve on Qdouble construction overheads;
  2. Homotopy solver;
  3. Migrate sparse linear solver from Csparse to UMFPACK;
- Interfaces:
  1. XML output of results
  2. XML input (data files) - Bug #55;
  3. SVG flowsheet connectivity output;
  4. Native Postgres database driver;
  5. Work flow for definition of a new model using XML: definition of the real, string and integer variables with automatic generation of results visualization schema, header file and maketables method - Bug #44, #54;
- Systems:



1. Support installing UI in Windows without administrative privileges and UAC (User Account Control) compatibility in Windows Vista
2. Match the requirements for the «Works with Windows Vista Logo Program for Software», the «Designed for Microsoft Windows XP» Application Specification and the «Application Specification for Microsoft Windows 2000 for Desktop Applications»
3. Install and update calculations kernels (as subproducts) independent from User Interface component;

### **Version 1.0**

Estimated deadline: 2009/08/30

Key deliverables:

- User interface:
  4. Problem configuration editing: will allow changing the values of selected string variables - requires smartenums support in UI;
  5. What-if and sensitivity analysis UI; allow to select list and ranges of variables, get results in tensor form, generate plots;
- Modelling:
  3. Expand selection of dynamic models;
  4. Tensor properties support à la SolidSim and population balance based phase model - Bug #6;
  5. Improve HTU/NTU based separation unit, non-equilibrium flash;
- Physical properties:
  4. Native UNIQUAC activity coefficient model;
  5. Native UNIFAC group-contribution predictive activity coefficient model;
  6. Natively equation of state PC-SAFT implementation - Bug #41;
  7. Interface to the VLXE SDK;
- Resolution:
  4. SQP minimization solver;
  5. Interface with BzzMath solvers;
- Interfaces:



6. Improve OPC interface;
  7. Provide calculation kernel-side scripting interface;
  8. Provide ASP or PHP component for remote web access, based on RPC (Remote Procedure Call) mechanism to the calculation kernel-side scripting interface;
  9. Work flow for problem structure editing: user will change the connectivity by editing a diagram, extract the structure of the problem and generate C++ code using scripting;
  10. CAPE-OPEN interface for plugging in external thermo models;
- Systems:
    4. Implement licensing i.e. web activation;
    5. Integrate with Plone CMS;