

NanoTrack – Sistema de Gerenciamento de Dados de Nanoestruturas com Plugins Inteligentes

André S. F. Silva, Paulo H. S. Batista, Artur M. M. Costa,
Alessandra C. Faria-Campos, Sérgio V. A. Campos and Omar P. Vilela Neto

¹Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG) Belo Horizonte – MG – Brazil

{andresfs, paulo.batista, artur.costa, scampos, alessa, omar}@dcc.ufmg.br

Abstract. *This work describes a tool to support nanoscience and nanotechnology laboratories based on the concept of LIMS (Laboratory Information Management Systems). The tool consists of a workflow capable of storing and managing data related to the synthesis of nanostructures, resources for data sharing and intelligent systems to infer parameters to be used in future synthesis of new nanostructures.*

Resumo. *Este trabalho descreve uma ferramenta para apoio a laboratórios de nanociência e nanotecnologia baseada no conceito de LIMS (Laboratory Information Management Systems). A ferramenta consiste de um workflow capaz de armazenar e gerenciar os dados relacionados à síntese de nanoestruturas além de recursos para o compartilhamento de dados e sistemas inteligentes para inferir parâmetros a serem utilizados em sínteses futuras de nanoestruturas.*

1. Introdução

O avanço tecnológico observado nas últimas décadas tem propiciado a geração, o armazenamento e a divulgação de uma grande quantidade de dados. Diversas áreas de pesquisa e desenvolvimento têm buscado alternativas eficientes para lidar com esta grande onda de informações, de forma a filtrar e trabalhar com os dados que são realmente essenciais. Entre estas áreas estão a nanociência e a nanotecnologia, que tratam do estudo e síntese de estruturas, dispositivos e sistemas controlando a forma e o tamanho na escala atômica e molecular. Recentemente, o interesse em nanociência e nanotecnologia tem aumentado devido ao grande potencial que esta área tem em propiciar benefícios em diversos campos, possibilitando o desenvolvimento de novos materiais com propriedades aprimoradas e inovadoras.

Para a fabricação de qualquer composto é essencial o conhecimento sobre o material que está sendo construído e em compostos nanométricos não é diferente. Porém, a obtenção destes dados é dificultada por vários fatores como o armazenamento precário de experimentos anteriores ou restrição de acesso aos dados de outros grupos, o que impede a descoberta de conhecimento pelo pesquisador e dificulta a criação de nanomateriais inovadores. Neste contexto, as ferramentas computacionais podem exercer um importante papel no auxílio às novas descobertas. Entretanto, a nanotecnologia ainda carece de ferramentas computacionais que apoiem o pesquisador em suas atividades [de la Iglesia et al. 2011].

Tendo em vista esta situação, o objetivo principal deste trabalho é construir um sistema computacional para gerenciamento de dados de experimentos em nanotecnologia com uma plataforma de apoio ao pesquisador que possua funcionalidades adequadas à captura e análise de dados desta natureza, como atividades de síntese, ambiente de simulação e compartilhamento de informação.

O sistema proposto é baseado em um LIMS (*Laboratory Information Management System*), um sistema desenvolvido com o objetivo de facilitar o gerenciamento e armazenamento de dados de laboratório que contempla todo o ciclo de vida dos dados (coleta de dados, armazenamento, análise, geração de relatórios e arquivamento) e permite o controle do acesso dos usuários aos resultados armazenados [Hinton 1995]. Além disso, o ambiente de simulação e descoberta de conhecimento proposto é constituído por sistemas de inferências de dados, tais como Redes Neurais Artificiais (RNA), que são encapsulados em módulos complementares e incorporados à estrutura do LIMS como *plugins*. O encapsulamento de cada simulador como *plugin* permitirá a evolução de cada sistema de inferência independente da evolução do LIMS, além de permitir que outros simuladores sejam integrados a ferramenta posteriormente.

O novo sistema, denominado *NanoTrack*, supre as principais necessidades de gerenciamento de dados de laboratórios de nanotecnologia, provendo uma ferramenta eficiente e segura para a utilização por pesquisadores da área. Com a construção deste sistema é possível uma melhora na qualidade e velocidade em que os dados são armazenados, recuperados e até mesmo analisados.

2. Trabalhos relacionados

Vários algoritmos e ferramentas computacionais para o gerenciamento de dados de laboratórios existem atualmente, como os LIMS desenvolvidos para outras áreas que podem ser vistos em [Ludäscher et al. 2006], [Oinn et al. 2006] e [Melo et al. 2010]. Entretanto, sistemas destinados ao uso por laboratórios de nanotecnologia são raros e pouco flexíveis [de la Iglesia et al. 2011]. Diante do avanço constante e rápido da área, projetar um *software* capaz de acompanhar estes avanços constitui um desafio. Normalmente, os LIMS são desenvolvidos para um laboratório específico, sendo que sua estrutura é pouco flexível à alterações. Além disto, a maioria possui código fonte fechado e um alto custo. [Informatics 2013] e [Prog4biz 2013] são empresas que comercializam *softwares* para gerenciamento de informações de laboratórios de nanotecnologia. Entretanto, estes não integram nenhum tipo de ferramenta de simulação. Por outro lado, a utilização de algoritmos de inferência já vem sendo aplicado à síntese de nanoestruturas. Existem atualmente diversos trabalhos disponíveis com algoritmos capazes de resolver problemas específicos, como [Singulani et al. 2008], [Nateri et al. 2011], [Cupertino et al. 2011] e [Dias et al. 2011]. Os trabalhos de [Wermelinger et al. 2008] e [Crnkovic et al. 2002] exploraram os princípios dos projetos conhecidos por impactar na manutenção de *software* e planejaram um modelo de modularização de ferramenta baseado nestes princípios para permitir a evolução da ferramenta através de *plugins*.

3. Nanotrack

Visando suprir a demanda de sistemas computacionais capazes de atender os laboratórios de nanotecnologia, neste trabalho foi desenvolvida a ferramenta *NanoTrack*, uma ferramenta orientada a workflows que utiliza como base o sistema Flux®. Este sistema é um

LIMS desenvolvido em Java com SGBD MySQL e interface *WEB* para acesso amigável. A ferramenta apresenta sistemas inteligentes de inferência e previsão integrados ao *workflow* como *plugins* que permitirão a aplicação de técnicas específicas de inferência quando necessário para auxiliar os especialistas na síntese de futuras nanoestruturas.

3.1. Modelagem

Cada laboratório possui características próprias de funcionamento e estas devem ser representadas em um modelo de dados a ser lido pela ferramenta. O primeiro passo para realizar a modelagem do laboratório é um entendimento da rotina de atividades do mesmo, com todas as suas etapas. Esta especificação dos requisitos é feita com auxílio de um pesquisador do laboratório a ser modelado que tenha conhecimento de toda a estrutura e funcionamento do mesmo. A rotina do laboratório é então modelada em forma de um fluxo de trabalho (*workflow*), onde as etapas identificadas dentro dos processos existentes nos laboratórios são representadas como atividades. Para cada dado associado a uma atividade existe um atributo correspondente dentro do sistema. Nesta etapa, materiais de apoio utilizados durante os experimentos também podem ser identificados, visando automatizar todo o processo, a fim de centralizar as informações num mesmo ambiente computacional. A seguir, toda a informação obtida durante as interações da especificação de requisitos é codificado no formato XPDL (*XML Process Definition Language*) [Interface 2002] para ser interpretada pelo *NanoTrack*. Para esta codificação foi utilizado o *Together Workflow Editor* [Teamsolutions 2011].

3.2. Plugins

Para integração dos sistemas de inferência ao *NanoTrack* foram utilizados *plugins*. O *plugin* é um módulo de extensão capaz de adicionar funções a uma ferramenta hospedeira. Os *plugins* são acoplados ao sistema a fim de prover funcionalidades específicas que não eram atendidas pela ferramenta, tornando o desenvolvimento destes componentes independente do desenvolvimento da aplicação. A Figura 1 mostra uma visão geral da estrutura e modelagem da ferramenta, demonstrando como é a interação dos *plugins*, como módulos independentes, com o *NanoTrack*. Nesta figura é possível observar que eles podem compartilhar a mesma base de dados utilizada pelo *NanoTrack*, permitindo assim que os dados armazenado pelo LIMS sejam utilizados pelos *plugins* para que estes possam realizar os processamentos necessários em suas operações.

3.3. Interação: Plugin x NanoTrack

Para que um *plugin* seja introduzido na aplicação deve ser criada uma nova atividade no *workflow*, contendo as informações que serão passadas para o processamento do *plugin*. Uma vez que os atributos das atividades são preenchidos e o procedimento é confirmado pelo usuário, a aplicação cria um processo para a execução do *plugin*. O funcionamento do *plugin* não importa para o *NanoTrack*. Para a aplicação ele funciona como uma caixa preta, onde os dados de entrada são inseridos e é esperada uma saída. A saída obtida durante a execução do *plugin* é armazenada em um arquivo texto, que foi especificado como parâmetro pelo *NanoTrack*. Este arquivo texto é posteriormente utilizado para exibir o resultados encontrados para o usuário da aplicação.

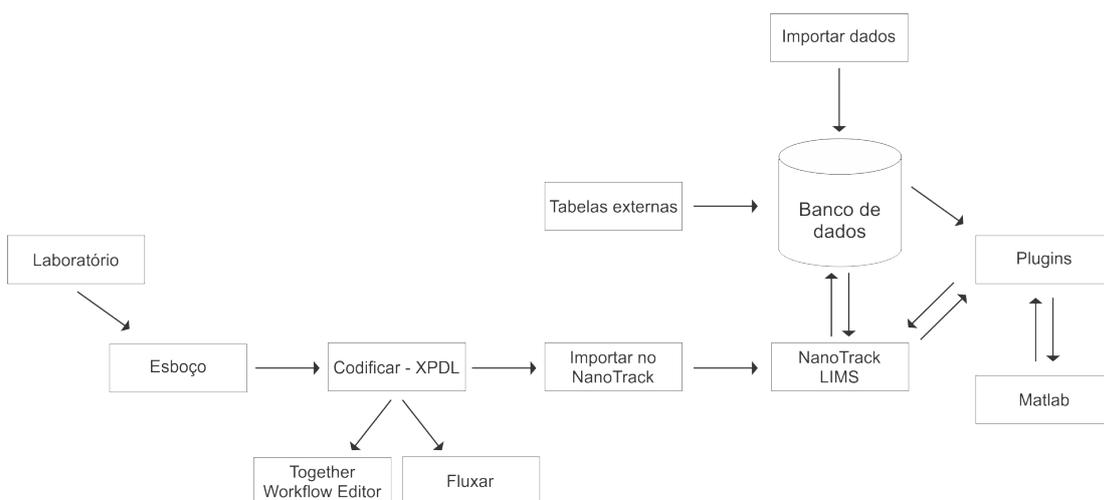


Figura 1. Visão geral da estrutura do NanoTrack.

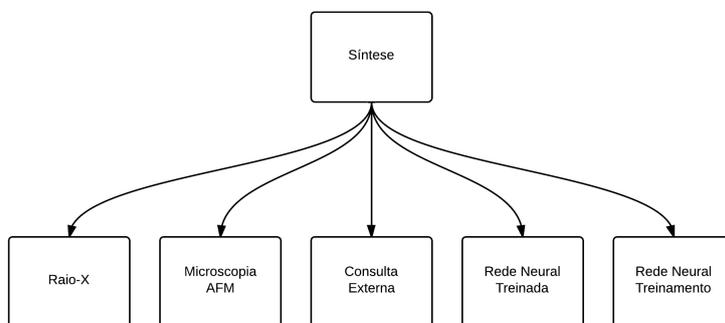


Figura 2. Workflow de experimentação de Pontos Quânticos.

4. Estudo de Casos

Neste projeto foram desenvolvidos três estudos de casos de laboratórios de nanotecnologia: **Pontos Quânticos**, **LabSem - Laboratório de Semicondutores (PUC-Rio)** e **Nanotubos de Carbono (Física-UFMG)**, sendo que, por limitação de espaço, apenas o primeiro será descrito neste trabalho.

4.1. Workflow do Laboratório de Síntese de Pontos Quânticos

A modelagem deste *workflow* foi baseada na experiência anterior em utilização de dados de Pontos Quânticos [Singulani et al. 2008], onde as etapas do processo foram identificadas e organizadas de forma ordenada. A Figura 2 mostra a estrutura do *workflow* com as diversas etapas. Estas etapas podem ser classificadas em dois grupos: síntese (Síntese) e caracterização (Raio-X e Microscopia AFM). Além destes, um terceiro grupo de atividades foi inserido formando a parte dos módulos incorporados à ferramenta, os *plugins* (Consulta Externa, Rede Neural Treinada e Rede Neural Treinamento).

Para cada processo identificado foi criada uma atividade que o representasse. Estas atividades são compostas por propriedades que representam os dados gerados durante a sua execução e estas deverão ser armazenados no banco de dados da ferramenta a fim de possibilitar futuras consultas e auxiliar os algoritmos de inferência na descoberta de conhecimento. Estas propriedades foram obtidas por meio do trabalho

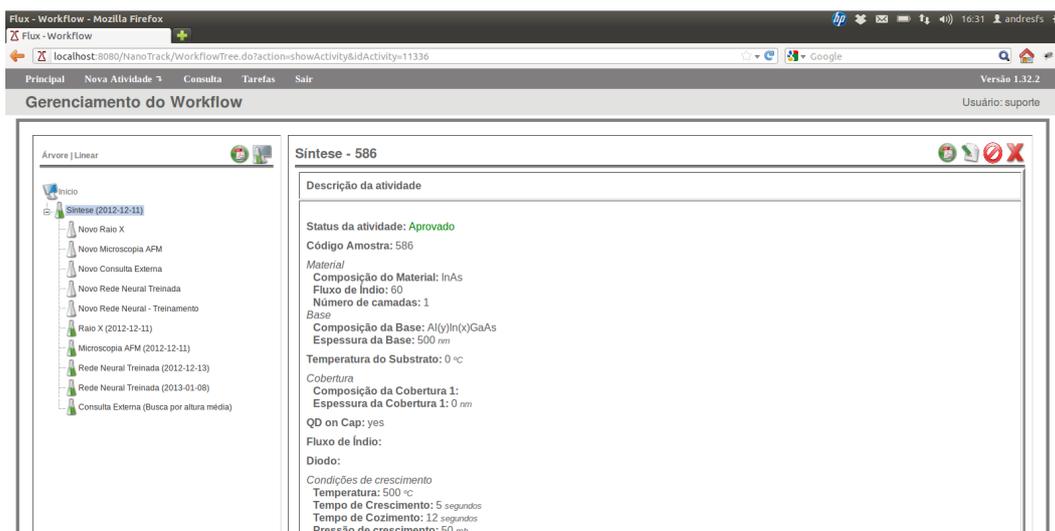


Figura 3. Interface do NanoTrack na atividade de síntese.

[Singulani et al. 2008], através de uma planilha eletrônica contendo um cadastro de diversos experimentos sobre a síntese de pontos quânticos.

Após modelar e identificar as propriedades do processo de experimentação de Pontos Quânticos é preciso codificá-lo para o formato XPDML que será interpretado pelo *NanoTrack*. Para isto a ferramenta *Together Workflow Editor (TWE)* foi utilizada. Durante a criação das atividades no TWE, as propriedades identificadas anteriormente são codificadas em forma de atributos.

A Figura 3 exibe a interface da atividade de síntese carregada através do *NanoTrack* com os dados de uma amostra sendo exibidos no lado direito da imagem. Além destes dados, é possível visualizar do lado esquerdo a estrutura do *workflow* desenvolvido e as demais atividades prontas para serem acessadas ou criadas.

A atividade “Consulta Externa” implementa um *plugin* que faz a busca por amostras cadastradas no *NanoTrack* que possuam os valores das propriedades desejadas, considerando um desvio. As amostras dentro do intervalo definido são apresentadas ao usuário de forma ordenada, priorizando aquelas mais próximas do valor desejado.

Neste *workflow* o ambiente de simulação e descoberta de conhecimento é composto por dois *plugins*, um para *treinamento da RNA* e outro que é a *RNA treinada*, ambos implementados em *Matlab*. Através da utilização do simulador, o usuário da ferramenta poderá ajustar os parâmetros de entrada à sua maneira e o simulador irá processar estes dados e retornar o resultado. Neste simulador foram implementados três redes neurais capazes de inferir informações sobre a experimentação em *Pontos Quânticos*, tais como: densidade, altura média e desvio padrão.

4.2. Plugin de Treinamento da Rede Neural Artificial

Este *plugin* é utilizado para treinar uma RNA do tipo *Perceptron* de Múltipla Camadas. As amostras que serão utilizadas no treinamento são previamente escolhidas durante o seu cadastro na ferramenta e, com base nesta informação, o *plugin* irá gerar um arquivo que servirá de suporte para o treinamento da RNA. Para a realização do treinamento por meio

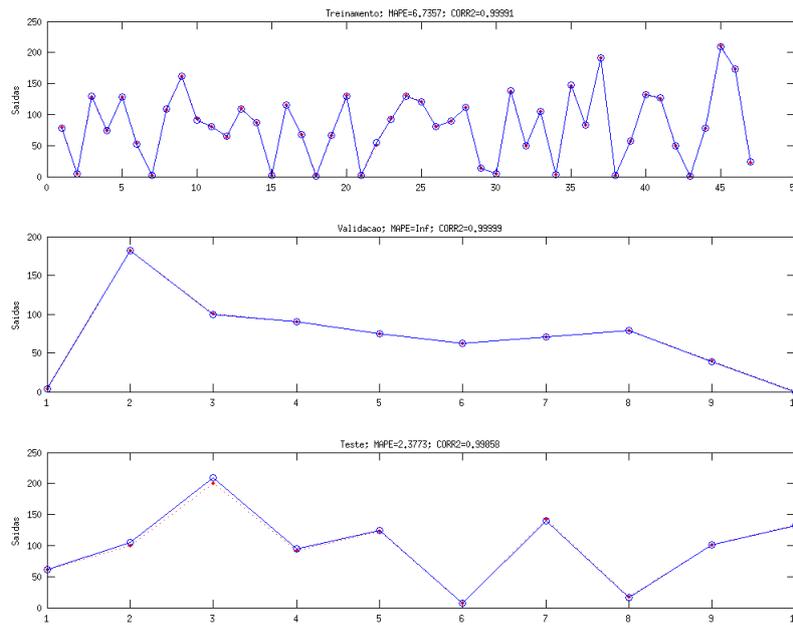


Figura 4. Resultado da RNA nas etapas de treinamento, validação e teste para a saída de Densidade.

do *NanoTrack*, a respectiva atividade é selecionada e seus atributos são passados para o *plugin* em forma de parâmetros.

Neste estudo de caso, três redes neurais foram treinadas para inferir a **densidade**, a **altura média** e o **desvio padrão** do processo de crescimento de pontos quânticos.

Os gráficos referentes à Rede que infere a **densidade** são apresentados na Figura 4. Os gráficos se referem às etapas de treinamento, validação e teste, respectivamente. A linha de cor vermelha representa o valor obtido pela rede neural durante a simulação e a de cor azul representa o valor esperado. A rede obteve resultados com altíssima precisão, sendo que o erro percentual absoluto médio (*MAPE - Mean Absolute Percentage Error*) [Armstrong and Collopy 1992] durante a etapa de teste foi de aproximadamente 2% e o coeficiente de correlação é próximo de 1, o que indica que o simulador é capaz de gerar resultados muito próximo do mundo real.

Os resultados da rede neural desenvolvida para a inferência de **altura média** foram satisfatórios com um erro médio de aproximadamente 11% durante a etapa de teste e correlação de 0,97. Já os resultados da rede neural para inferência de **desvio padrão** não foram satisfatórios, apresentando um erro médio de quase 30% e uma correlação de 0,15. A RNA para inferência de desvio padrão não foi aprimorada devido a falta de precisão das amostras cadastradas, entretanto o objetivo maior era mostrar a capacidade de integração do ambiente de simulação com o LIMS.

4.3. Plugin de Uso da Rede Neural Artificial

Nesta etapa, a melhor rede neural obtida no passo anterior, que será utilizada como simulador, é carregada na memória do *Matlab*. Com isto, todo o ambiente que foi gerado durante a criação da rede neural fica disponível. Para utilizar a rede neural o usuário deverá informar os valores de entrada que ele deseja simular. A atividade então invoca o



Figura 5. Utilização da RNA treinada no ambiente de simulação do *NanoTrack*.

plugin. Como os dados utilizados durante o treinamento foram normalizados, é importante que os dados a serem simulados também sejam normalizados. Após a normalização dos dados, a Rede Neural é invocada e os valores de entrada, são passados por parâmetro. A inferência é realizada de maneira extremamente rápida e o resultado obtido pela rede neural fica disponível, bastando apenas realizar o processo inverso da normalização para que os valores obtidos sejam condizentes com os valores reais. A Figura 5 exibe um exemplo de simulação realizado através do *NanoTrack*. Na figura, os atributos utilizados como entrada na rede neural estão listados na tabela e, após a tabela, o resultado obtido para estes parâmetros pode ser visto no campo “Resultado da simulação”.

5. Conclusão

Como discutido, a nanotecnologia carece de ferramentas computacionais capazes de apoiar o pesquisador em suas atividades. Por isto, a gestão das informações geradas pelos laboratórios é prejudicada, impedindo um melhor aproveitamento dos dados e dificultando a descoberta de conhecimento. Com o objetivo de suprir estas carências, neste trabalho foi desenvolvido o *NanoTrack*, um LIMS, baseado na plataforma Flux®®, responsável por gerenciar os experimentos realizados em um laboratório de nanotecnologia, a fim de apoiar a criação de nanomateriais. Além do gerenciamento dos dados, foi mostrado que o *NanoTrack* pode incorporar novas funcionalidades através da utilização de *plugins*. Através desta característica foi possível criar um ambiente de simulação formado por sistemas de inferência encapsulados em *plugins*. Além dos simuladores desenvolvidos, também foi criado um algoritmo de busca alternativa. Esta busca é caracterizada por suportar margens de erros e que não é atendida pelos algoritmos convencionais. Através dela, resultados que poderiam satisfazer a necessidade do pesquisador e não eram obtidos passam a ser mostrados. O desenvolvimento do sistema *NanoTrack* apresenta uma solução que poderá contribuir significativamente para a pesquisa em nanotecnologia ao auxiliar na organização e análise dos dados obtidos e propor sistemas de inferência automatizados que facilitarão a seleção dos dados para análise.

Referências

- Armstrong, J. S. and Collopy, F. (1992). Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting*, 8(1):69–80.
- Crnkovic, I., Larsson, S., and Stafford, J. (2002). Component-based software engineering: building systems from components at 9th iee conference and workshops on engineering of computer-based systems. *ACM SIGSOFT Software Engineering Notes*, 27(3):47–50.
- Cupertino, L., Vilela Neto, O., Pacheco, M., Vellasco, M., and d’Almeida, J. (2011). Modeling the young modulus of nanocomposites: A neural network approach. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 1599–1605. IEEE.
- de la Iglesia, D., Harper, S., Hoover, M., Klaessig, F., Lippell, P., Maddux, B., Morse, J., Nel, A., Rajan, K., Reznik-Zellen, R., et al. (2011). Nanoinformatics 2020 roadmap.
- Dias, D. M., Singulani, A. P., PACHECO, M. A. C., SOUZA, P. L., Pires, M., and Vilela Neto, O. P. (2011). Self-assembly quantum dots growth prediction by quantum-inspired linear genetic programming. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pages 2075–2082. IEEE.
- Hinton, M. (1995). Laboratory management systems. *New York*.
- Informatics, C. (2013). Core lims.
- Interface, P. (2002). Xml process definition language. *Document Number WPMC-TC-1025 Document Status-XPDL*, 1.
- Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E., Tao, J., and Zhao, Y. (2006). Scientific workflow management and the kepler system. *Concurrency and Computation: Practice and Experience*, 18(10):1039–1065.
- Melo, A., Faria-Campos, A., DeLaat, D. M., Keller, R., Abreu, V., and Campos, S. (2010). Sigla: an adaptable lims for multiple laboratories. *BMC genomics*, 11(Suppl 5):S8.
- Nateri, A., Dadvar, S., Oroumei, A., and Ekrami, E. (2011). Prediction of silver nanoparticles diameter synthesized through the tollens process by using artificial neural networks. *Journal of Computational and Theoretical Nanoscience*, 8(4):713–716.
- Oinn, T., Greenwood, M., Addis, M., Alpdemir, M., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., et al. (2006). Taverna: lessons in creating a workflow environment for the life sciences. *Concurrency and Computation: Practice and Experience*, 18(10):1067–1100.
- Prog4biz (2013). Bookitlab.
- Singulani, A., Vilela Neto, O., Aurélio Pacheco, M., Vellasco, M., Pires, M., and Souza, P. (2008). Computational intelligence applied to the growth of quantum dots. *Journal of Crystal Growth*, 310(23):5063–5065.
- Teamsolutions, T. (2011). Together workflow editor.
- Wermelinger, M., Yu, Y., and Lozano, A. (2008). Design principles in architectural evolution: a case study. In *Software Maintenance, 2008. ICSM 2008. IEEE International Conference on*, pages 396–405. IEEE.