

## Proposta de arquitetura modular para software de planejamento de processo para impressão 3D

### RESUMO

O software RP3 foi idealizado para realizar o planejamento de processo de tecnologias de Manufatura Aditiva. Algumas das etapas necessárias no planejamento são: o cálculo do fatiamento, do preenchimento, da definição de suporte, otimizações, entre outros. Devido a sua natureza científica, que visa a realização de pesquisa, o RP3 deve suportar mudanças constantes de métodos em cada uma destas etapas, bem como, deve permitir comparar diferentes métodos para a mesma etapa. Existem várias ferramentas que prometem resolver o problema da modularização na literatura de Engenharia de Software, como os plugins, Design Patterns, princípios de design de software como SOLID, GRASP, etc. Neste trabalho foi proposto aplicar alguns desses conceitos no RP3, sendo então discutidos as justificativas de cada implementação, o custo técnico da implementação, as dificuldades e os benefícios de se implementar bons padrões de projeto em um sistema altamente dinâmico como o RP3.

**PALAVRAS-CHAVE:** Padrões de Projeto. Engenharia de *Software*. Manufatura Aditiva.

**Gustavo Yudi Bientinezi Matsuzake**  
[matsuzake@alunos.utfpr.edu.br](mailto:matsuzake@alunos.utfpr.edu.br)  
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

**Neri Volpato**  
[nvolpato@utfpr.edu.br](mailto:nvolpato@utfpr.edu.br)  
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

**Rodrigo Minetto**  
[rodrigo.minetto@gmail.com](mailto:rodrigo.minetto@gmail.com)  
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

## 1 Introdução

O RP3 é um *software* para planejamento de processo de manufatura aditiva. De uma maneira geral, o planejamento de processo é dividido em etapas como: abrir uma geometria 3D de um objeto (arquivo .stl, por exemplo); posicionar a geometria na bandeja da máquina; fatiar a geometria; criação de suportes de sustentação (se necessário); definição de preenchimento dos contornos de cada fatia; geração de código de máquina; entre outros. Existem vários princípios de tecnologias de manufatura aditiva, cada qual com suas necessidades peculiares de processo. Se aprofundar no processo de manufatura aditiva em si, ou em cada uma das suas etapas não é o propósito desse trabalho, e sim, focar na arquitetura do RP3 e na realização dessas tarefas de forma flexível, onde cada uma de suas etapas é facilmente alterada, substituída ou removida sem prejudicar as funcionalidades do resto do sistema.

O RP3 foi criado dentro de um contexto acadêmico, logo, sua natureza científica demanda uma certa flexibilidade, uma vez que todas as etapas do processo de manufatura é um ramo de pesquisa por si só. Então, para acompanhar o estado da arte e realizar novas contribuições científicas o RP3 tem os seguintes requisitos:

1. O *software* deve ter uma independência da tecnologia de manufatura aditiva;
2. O *software* deve ser modularizado onde cada etapa possa ser alterada, substituída e removida facilmente.

## 2 Métodos

Através de uma análise da arquitetura do RP3 foram encontrados três problemas do ponto de vista dos requisitos do sistema.

O primeiro (chamado também de Problema 1) é a não idealização do sistema em uma arquitetura *MVC* (**M**odel-**V**iew-**C**ontroler) (REENSKAUG, 1979). Todos os componentes gráficos eram criados de forma “manual” e estática dentro do código, criando a necessidade de localizar as linhas do código para alterar um recurso visual. Manutenções dessa forma são muito suscetíveis a erros, uma vez que a interface e a lógica do RP3 estavam de certa forma relacionados.

O segundo (Problema 2) é a implementação do RP3 visando apenas uma tecnologia de manufatura aditiva (a extrusão de material), não havendo uma modularização das etapas do processo. Isso dificultava a implementação do requisito 1. As hierarquias dos objetos que compunham as etapas não foram implementadas idealizando o requisito 1 e 2.

O terceiro (Problema 3) é que as etapas não eram contidas em si, i.e., cada nova etapa a ser projetada e/ou incluída, deveria também, mudar instâncias internas do RP3, e isso torna o sistema muito mais suscetível a erros (FOWLER, 2011).

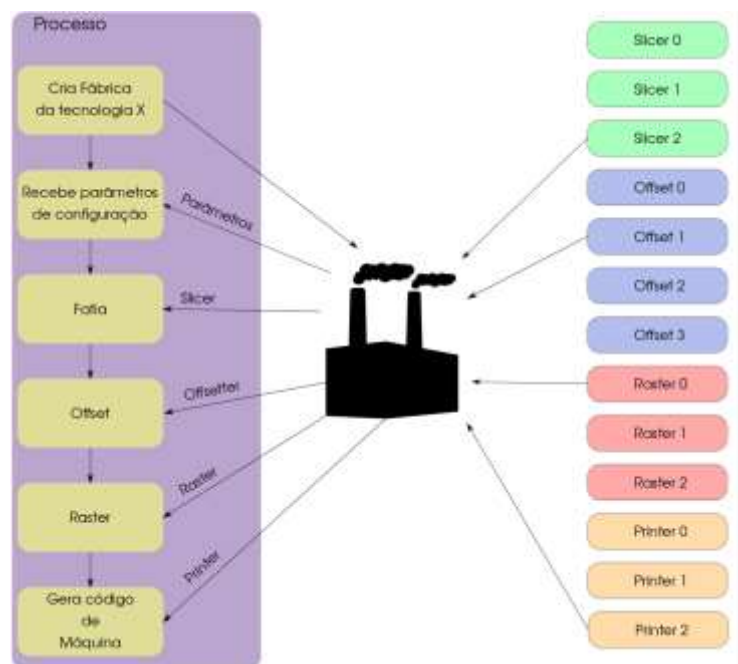
Foram propostas três soluções para esses três problemas. Sendo a solução para o Problema 1 uma solução parcial. Cada solução foi avaliada com a satisfação dos requisitos 1 e 2.

A solução proposta para o Problema 1 foi a utilização de uma das ferramentas implementadas no *framework Qt*, utilizado para a interação com o usuário no RP3. Essa solução foi escolhida pois, como a ferramenta uic já está contida no *framework Qt*, então, não foi necessária nenhuma alteração no ambiente de desenvolvimento. Essa tradução de código foi feita botão à botão, janela a janela, *widget a widget*.

A solução para o Problema 2 foi a utilização de um conceito muito disseminado em padrões de projeto: as fábricas abstratas. Fábricas são objetos dentro do *software* que instanciam (fabricam) outros objetos e fábricas abstratas são um modelo de como e quais objetos essas fábricas devem fabricar. São importantíssimos para a modularização e interfaceamento de módulos (LASATER, 2006).

A ideia de fábrica implementada no RP3 é a ideia de uma “fábrica da tecnologia”, onde cada tecnologia terá sua fábrica e suas “matérias-primas”, que nesse contexto, seriam as etapas. Uma ilustração dessa arquitetura pode ser vista na Figura 1.

Figura 1: Ilustração do conceito de fábrica abstrata.



Fonte: Autoria própria.

A solução do Problema 3 foi visando sanar a dependência entre as etapas e a estrutura interna do sistema. Essa dependência estava relacionada aos parâmetros de configuração de cada etapa, que eram feitos de maneira estática. Como as etapas são dinâmicas, os parâmetros para essas etapas também serão dinâmicos. Então foi implementado um conceito de registro de parâmetros, padrão esse que já foi utilizado em sistemas antes, e é muito difundido, inclusive, o *Microsoft Windows* usa um conceito semelhante (FOWLER, 2011).

---

## 2 Resultados e discussão

Após a implementação das soluções, a interface e a lógica de manufatura aditiva estavam completamente independentes. Os benefícios dessa separação é muito importante para a independência dos módulos, agora, um *designer* pode trabalhar de forma independente de um programador no RP3 utilizando o Qt *Designer*. O Problema 1 foi resolvido parcialmente com essa solução porque idealmente as visões, os modelos e o controlador deveriam ser totalmente separados. Com essa solução apenas as visões estão separadas, o controlador e os modelos ainda estão dependentes no sistema.

Com o conceito de fábrica abstrata, cada etapa do processo pode ser modificados facilmente, uma vez que é alterado a “matéria-prima” da fábrica. É importante notar também que como a fábrica é uma classe base e abstrata, a própria fábrica pode ser alterada facilmente, podendo assim, alterar a tecnologia. Idealmente, existiria para cada tecnologia uma fábrica: fábrica de extrusão, fábrica de fotopolimerização e entre outras.

Com o padrão de registros, cada etapa do RP3 está independente. Na implementação de um módulo não há necessidade de buscar ou modificar o núcleo do sistema.

Apesar dos benefícios das propostas aqui apresentadas, a arquitetura do RP3 ficou relativamente mais complexa. Novos integrantes que trabalharão com o núcleo do sistema terão uma curva de aprendizagem do sistema um pouco maior. Porém, ao se analisar o sistema como um todo e os benefícios da solução, isso acaba se tornando aceitável, uma vez que sem soluções semelhantes, o RP3 não conseguiria alcançar seus requisitos.

## 3 Conclusões

As soluções propostas para o problema de arquitetura de software do RP3 tornam os requisitos 1 e 2 possíveis de serem realizados. Além disso, facilitam a manutenção do código e como cada módulo ficou dependente apenas de si, várias pesquisas, em etapas diferentes no processo de manufatura aditiva, podem ser feitas simultaneamente. Como sugestões para trabalhos futuros ficam terminar a arquitetura MVC e modularização na parte interior das etapas.

---

## Modular architecture for 3D printing process planning software

### ABSTRACT

The RP3 software was designed to assist in the process planning for additive manufacturing technologies. Some of the steps needed in process planning are: slicing, filling, support definition, optimizations, and more. Due to its scientific nature, the RP3 must support constant changes of methods in each of these steps and it should allow to compare different methods for the same step as well. Several tools and ideas that promise to solve the problem of modularization can be found in the literature, such as plugins, Design Patterns, principles of software design like SOLID, GRASP, and others. In this work we applied some of these concepts in RP3 and discussed justifications and technical cost of each implementation, difficulties and benefits of implementing good design patterns in a highly dynamic system such as RP3.

**KEYWORDS:** Additive Manufacturing. Design Patterns. Software Engineering.

## AGRADECIMENTOS

Agradecimentos especiais são direcionados ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) pelo incentivo financeiro ao projeto.

## REFERÊNCIAS

FOWLER, Martin. **Patterns of Enterprise Application Architecture**. Boston: Addison-wesley, 2011.

LASATER, Christopher G.. **Design Patterns**. Plano: Wordware Publishing, Inc., 2006.

QT. QT User Interface Compiler. 2016. <<http://doc.qt.io/qt-4.8/uic.html>>. [Online; acessado em 04/2017].

REENSKAUG, Trygve. THING-MODEL-VIEW-EDITOR: an Example from a Planning System. **Xerox Parc Technical Note**, Palo Alto, p.0-14, maio 1979.

**Recebido:** 31 ago. 2017.

**Aprovado:** 02 out. 2017.

**Como citar:**

MATSUZAKE, G. Y. B. et al. Proposta de arquitetura modular para software de planejamento de processo para impressão 3D. In: SEMINÁRIO DE INICIAÇÃO CIENTÍFICA E TECNOLÓGICA DA UTFPR, 22., 2017, Londrina. Disponível em: <<https://eventos.utfpr.edu.br/sicite/sicite2017/index>>. Londrina: UTFPR, 2017.

**Correspondência:**

Gustavo Yudi Bientínez Matsuzake  
Rua Professor Rubens Elke Braga, número 551, Bairro Parolin, Curitiba, Paraná, País.

**Direito autoral:**

Este resumo expandido está licenciado sob os termos da Licença Creative Commons-Atribuição-Não Comercial 4.0 Internacional.

