

# Apresentação de Atividade Remotas

Igor Pires dos Santos

[igor.pires@ice.ufjf.br](mailto:igor.pires@ice.ufjf.br)



Programa de Pós-Graduação em Modelagem Computacional  
Universidade Federal de Juiz de Fora

November 13, 2020

Objetivos

Modelagem

O Programa

Atividades Paralelas

Conclusão

A ferramenta IGU através de uma modelagem de classe simples ela pôde ser utilizada para visualizar diversas propriedades da árvore em tempo real enquanto à iterávamos com a metodologia proposta por Duan & Zamir.

Durante o mestrado pretendo remodelar e reescrever o código da ferramenta, à fim de padronizá-la e aumentar sua praticidade na hora de executar as simulações. Bem como aprofundar os conhecimentos nas simulações executadas pela ferramenta

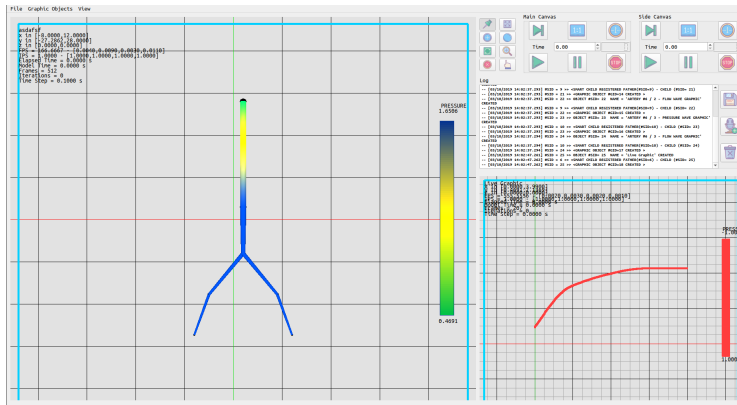


Figure 2: Interface IGU

Os principais objetivos desta nova ferramenta são:

- ▶ Armazenar todos os frames de uma animação.
- ▶ Possibilitar o carregamento de uma estrutura única através de um arquivo vtk ou xml.
- ▶ Gerenciar elementos para que sejam armazenados e carregados de forma correta.
- ▶ Permitir através de sua modelagem de classes que a mesma estrutura de dados seja manipulada por diferentes máquinas de iteração e gráficas.
- ▶ Preparar o programa para uma futura paralelização e distribuição.
- ▶ Definir ambiente de compilação através de uma imagem do Docker.
- ▶ Definir ambiente de continua integração e despache com Jenkins & Docker.

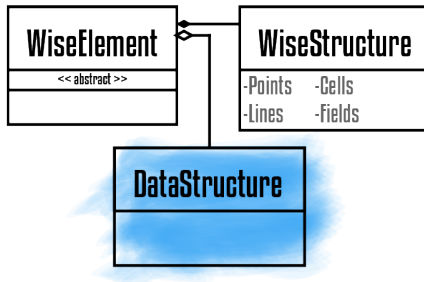


Figure 3: WiseElement

WiseElement  
Statuses

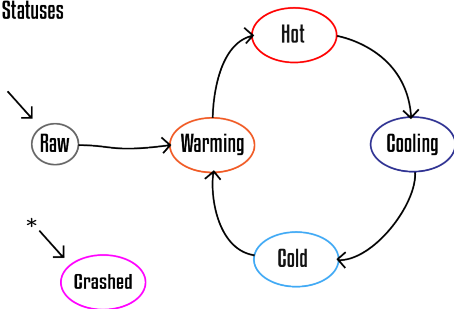


Figure 4: Status de WiseElement

## Points



Figure 5: Points

Os pontos armazenados estão contidos em  $R^3$ . É o elemento da estrutura mais simples, usa coordenadas  $(x, y, z)$ . Pontos unidimensionais ou bidimensionais utilizam os mesmos campos, zerando a(s) coordenada(s) não utilizada(s).



## Line

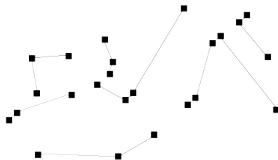


Figure 6: Lines

As linhas utilizam da estrutura pré-existente de pontos. Linhas podem ser interpretadas de formas diferentes para objetos diferentes e podem conter múltiplos pontos. Podem ser, por exemplo, arestas de um grafo ou a artéria de uma árvore arterial.

## Cell

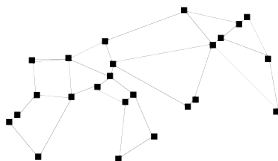


Figure 7: Cells

As células também utilizam da estrutura pré-existente de pontos. Células representam estruturas mais complexas, pois além da quantidade de pontos também possui o tipo da célula (int) que pode ser interpretado de formas diversas.

## Fields

Os campos aqui mencionados guardam os dados relacionados ao modelo inteiro, como por exemplo, tempo, de iterações, nome e outros. Cada "campo" possui um nome e um vetor que é traduzido ao *Data Type* desejado.

Esses campos possuem ainda mais 2 parâmetros, a família e o tipo de campo.

- ▶ Wise
- ▶ Iteration
- ▶ Graphic
- ▶ Initial
- ▶ Live
- ▶ Out

## Data & Datatype

Todos os campos anteriormente mencionados, com exceção do Fields possui uma matriz de dados guardada. Através do tipo da estrutura (ponto, linha ou célula), o nome do campo e o ID daquela estrutura é possível acessar diretamente a informação armazenada na matriz (*Data*) que pode ser convertido para o seu respectivo *DataType*. A matriz mencionada armazena todos os valores em *QString*.

- ▶ Int
- ▶ Float
- ▶ Double
- ▶ String
- ▶ Vector
- ▶ Tensor
- ▶ Unknown

## Status do WiseElement

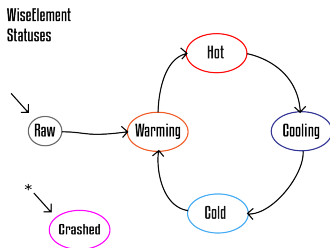


Figure 8: Status do WiseElement

## Status do WiseElement

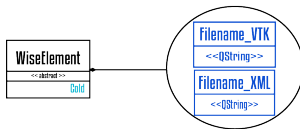


Figure 9: Cold WiseElement

## Status do WiseElement

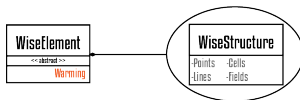


Figure 10: Warming/Cooling/Raw WiseElement

## Status do WiseElement

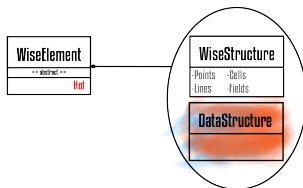


Figure 11: Hot WiseElement



## Tipos de WiseElement

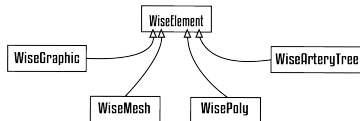


Figure 12: Tipos do WiseElement

## Fábricas

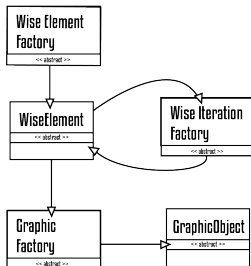


Figure 13: Conceito de Fábricas

## Fábricas

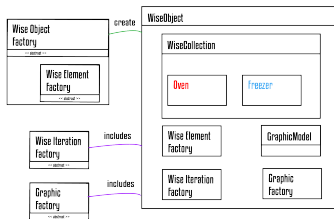


Figure 14: Conceito de Fábricas Re-imaginado

Como dito anteriormente, toda uma modelagem de classes foi criada para re-imaginar o programa. Entretanto, em paralelo está o repositório com o IGU 1.0, todas as simulações serão feitas na versão "antiga" até que a versão atualizada esteja completa.

Estima-se também que uma vez completa será possível abrir os mesmos modelos em ambas as ferramentas.

O código do IGU 2.0 já se encontra no repositório e está em desenvolvimento. Hoje, é possível abrir as estruturas em VTK e XML já com a estrutura de classes nova e iterá-lo manualmente. Entretanto a parte gráfica ainda não funciona.

Ainda neste periodo de Quarentena eu adquiri novos cursos sobre os assuntos que tenho contato no desenvolvimento da ferramenta.

- ▶ *Docker*
- ▶ *Jenkins*
- ▶ QT (**Iniciante** + Intermediário + Avançado)

Além disso, possuo um programa simples em Qt que serve de exemplo de exportação simples (Windows, Unix, MAC).

Outras atividades propostas pelo Rafael foram:

- ▶ Ler artigo CCO. **OK**
- ▶ Resumir artigo CCO.
- ▶ Conferir cálculos em árvores simples. **OK**
- ▶ Refazer modelos apresentados. **OK**

Nos próximos meses será terminado por completa a ferramenta e testada com os resultados obtidos previamente.