XVI EAMC - Towards Provenance Support in the BioinfoPortal Gateway

Marco Cabral[1,2], Antônio Tadeu Azevedo Gomes[1], Marcelo Galheigo[1], Kary Ocaña[1]
[1] Laboratório Nacional de Computação Científica (LNCC)
[2] Universidade Federal do Rio de Janeiro (UFRJ)

{macabral, atagomes, galheigo, karyann}@lncc.br

introdução

O *gateway* O gateway Bioinfo-Portal (https://bioinfo.lncc.br/) visa a execução de aplicações de bioinformática em larga escala, no apoio às pesquisas da comunidade científica de bioinformática. Bioinfo-Portal está acoplado a recursos de computação de alto desempenho (CAD) e do supercomputador Santos Dumont a fim de diminuir o tempo de processamento de execuções. Bioinfo-Portal gerencia a execução automática de aplicações, ferramentas e coleções de dados científicos através de uma interface web amigável e iterativa e das diversas camadas de software do *gateway*. Bioinfo-Portal utiliza, via serviços Web RESTful, o *middleware* CSGrid como *framework* de integração à arquitetura do SINAPAD. Atualizações e otimizações do Bioinfo-Portal na camada de banco de dados e de gerência de execuções irão fornecer uma melhor funcionalidade e escalabilidade de processos de execuções e armazenamento de dados de proveniência, tal que auxiliem na tomada de decisões inteligentes no uso de recursos computacionais.

Objetivos

- Atualização das camadas de banco de dados e de gerência de execuções da arquitetura do Bioinfo-Portal, por meio do desenvolvimento de serviços específicos para integrar dados contidos nessas camadas.
- Análise, extração e gerência de informações de dados científicos e de proveniência extraídas das camadas da arquitetura do Bioinfo-Portal e das aplicações de bioinformática.
- Implementação e validação de um banco de dados que centralize informações do Bioinfo-Portal e do ambiente computacional.
- Desenvolvimento de sistemas para criar inteligência em análise de coleta de dados e tomada de decisão, tal que melhore a eficiência do *gateway* em termos de velocidade, execução e armazenamento.

Metodologia

- Na primeira etapa, o projeto físico utilizou o PostgreSQL v10 como Sistema de Gerência de Banco de Dados (SGBD) relacional *Open Source* e *pgAdmin* v5.2 como plataforma de desenvolvimento e gerência.
- A segunda etapa envolve a utilização de serviços *RESTful* para o desenvolvimento dos sistemas de tomada de decisão inteligentes. A linguagem de programação utilizada é a PHP (*Hypertext Preprocessor*). Visual Studio Code é o editor de código-fonte usado para o desenvolvimento dos sistemas.

Referência

[1] Ocaña, K.A.C.S., et al. (2020). BioinfoPortal: A scientific gateway for integrating bioinformatics applications on the Brazilian national high-performance computing network. *In Future Generation Computer Systems*, Rio de Janeiro, v. 107, p. 23, Janeiro 2020.

[2] KIM, S.-H. et al. (2017). Science Gateway Cloud With Cost-Adaptive VM Management for Computational Science and Applications. *IEEE Systems Journal* , v. 11, n. 1, p. 173-185, Março 2017. ISSN 1932-8184.

[3] LESK, A. M (2019). Bioinformatics,*Britannica*, Pennsylvania, Fevereiro 2019.

[4] Gesing S, Krüger J, Grunzke R, Herres-Pawlis S, Hoffmann A. (2016). Using Science Gateways for Bridging the Differences between Research Infrastructures,*Journal of Grid Computing*, 2016;14:545–57.

Resultados I: Banco de dados

O modelo conceitual de banco de dados do Bioinfo-Portal foi implementado, como apresentado na Figura 1. Iniciou-se a o mapeamento dos dados na arquitetura do *gateway* para a implementação do modelo lógico.

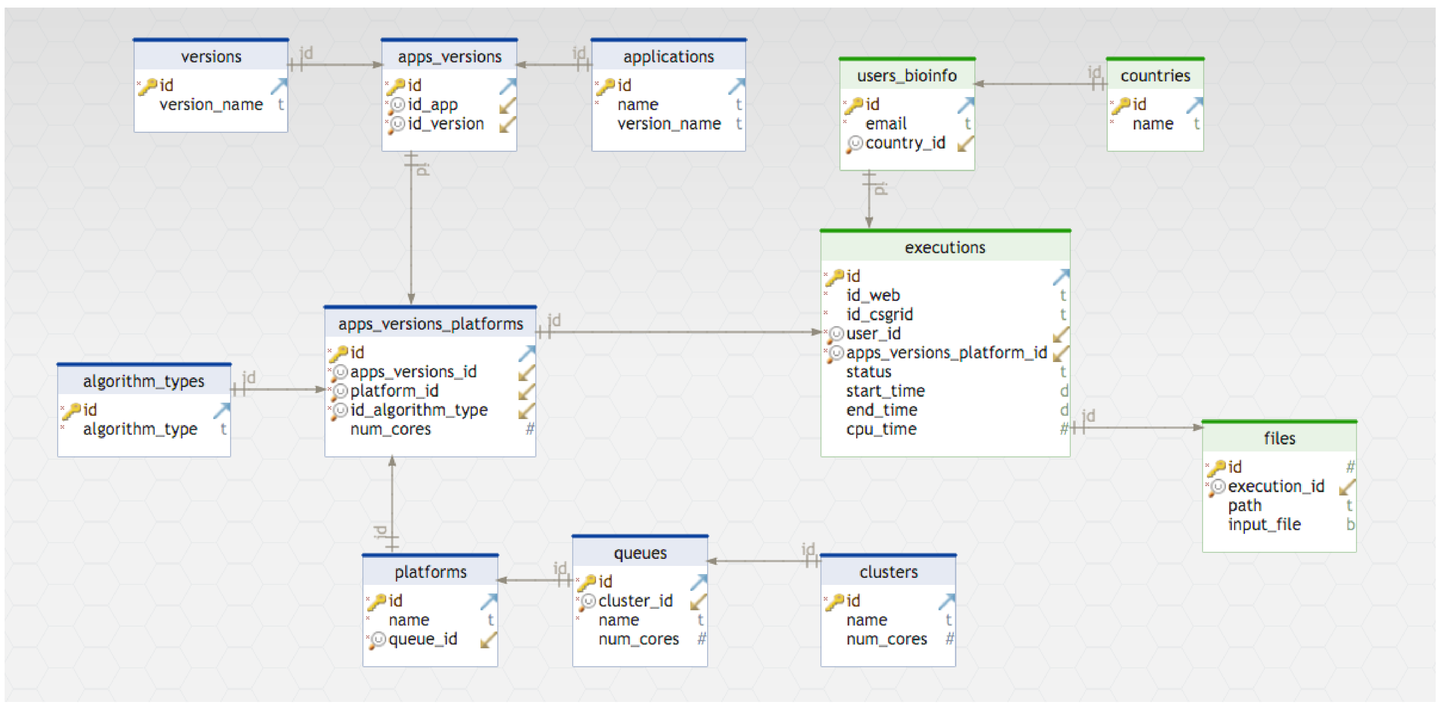

Figura 1. Esquema Conceitual Entidade-Relacionamento do Banco de Dados do Bioinfo.

Dentre as entidades do modelo conceitual ER do Bioinfo-Portal (Figura 1), *Files* e *Executions* são entidades originais, as demais entidades pertencem à nova versão do banco de dados.

Resultados II: Sistemas inteligentes

Em desenvolvimento, os sistemas utilizando serviços web *RESTful* visam interagir dinamicamente com o *middleware* CSGrid do SINAPAD. A Figura 2 apresenta o Sistema de Autenticação, por meio do método *LDAP* (Figura 2A) e *RSA* (Figura 2B). Esses sistemas extraem, tratam e armazenam dados de proveniência de usuários, como nome e identificação.

```php
<?php
    $url = ('http://ebio.sinapad.lncc.br:8888/rest/api/authentication/login-ldap');

    $data = array(
     'username'=> 'marco.antonio',
     'password'=> 'ant0nio',
     'service'=> 'ant0nio',
     'uuid' => $uuid

    );
    $headers = array(
     #'Accept: application/xml'
      'Accept: application/json'
    );

    $handle = curl_init();
    curl_setopt($handle, CURLOPT_URL, $url);
    curl_setopt($handle, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($handle, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($handle, CURLOPT_SSL_VERIFYHOST, false);
    curl_setopt($handle, CURLOPT_SSL_VERIFYPEER, false);

    curl_setopt($handle, CURLOPT_POST, true);
    curl_setopt($handle, CURLOPT_POSTFIELDS, http_build_query($data));

    $response = curl_exec($handle);
    $obj = json_decode ( $response );
    $b = $obj->{'uuid'};
    //echo $response;
?>
```

Figura 2A: Sistema de Autenticação *LDAP*

```php
<?php
$url = "http://ebio.sinapad.lncc.br:8888/rest/api/authentication/login-rsa";
    $data = array (
                    'service' => "CSGrid",
                    'username' => "BioInfoDiscoveryServices",
                    'file' => new CurlFile("/Users/marcoantonio/BioInfoDiscoveryServices.key", 'multipart/form-data')
    );
    $headers = array (
                    'Accept: application/json'
    );
    $handle = curl_init ();

    curl_setopt ( $handle, CURLOPT_URL, $url );
    curl_setopt ( $handle, CURLOPT_HTTPHEADER, $headers );
    curl_setopt ( $handle, CURLOPT_RETURNTRANSFER, true );
    curl_setopt ( $handle, CURLOPT_SSL_VERIFYHOST, false );
    curl_setopt ( $handle, CURLOPT_SSL_VERIFYPEER, false );
    curl_setopt ( $handle, CURLOPT_CONNECTTIMEOUT, 120 );
    curl_setopt ( $handle, CURLOPT_TIMEOUT, 600 );
    curl_setopt ( $handle, CURLOPT_POST, true );
    curl_setopt ( $handle, CURLOPT_POSTFIELDS, $data );
    curl_setopt ( $handle, CURLINFO_HEADER_OUT, true);

    $response = curl_exec ( $handle );

$obj = json_decode ( $response );
$b = $obj->{'uuid'};

//echo $response;
?>
```

Figura 2B: Sistema de autenticação *RSA*

Conclusão

A implementação dos serviços propostos e da base de dados atualizada permite melhorias no desempenho e funcionalidade do BioinfoPortal. BioinfoPortal é mais eficiente em termos de armazenamento, velocidade e funcionalidade para gerenciamento de arquivos e envio de trabalhos Como próximo passo, o aprendizado de máquina será acoplado como soluções em análise preditiva. Além disso, o desenvolvimento de sistemas para mapear os dados de localização do usuário (IP, País) e para mapear as informações dos dados de envio