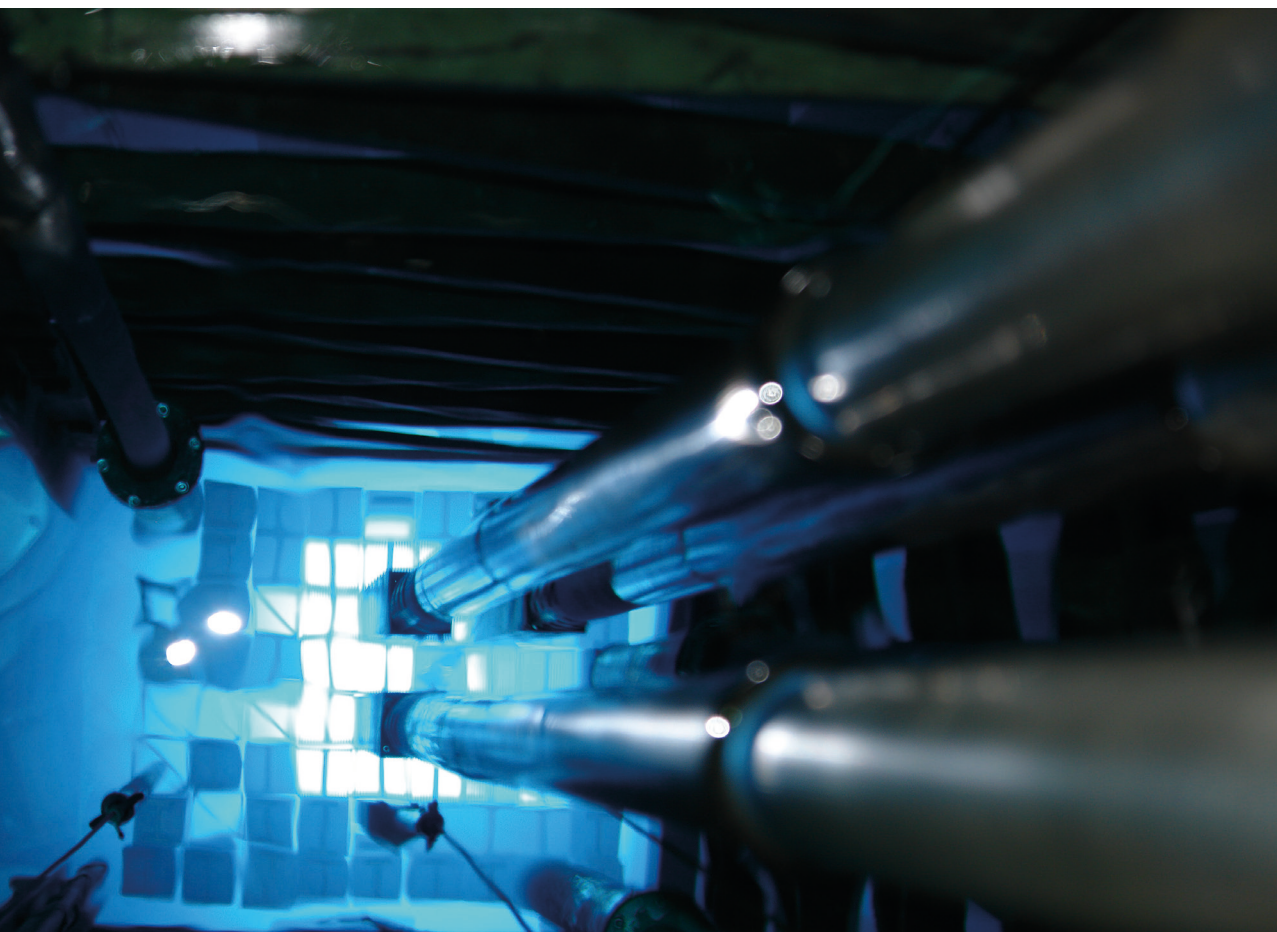


Paulo Sergio Cardoso Da Silva
Guilherme Soares Zahn
Francisco De Assis Souza
organizadores

CONTRIBUIÇÕES DO REATOR IEA-R1 PARA A PESQUISA NUCLEAR

WARP2: II Workshop Anual do Reator de Pesquisas



CONTRIBUIÇÕES DO
REATOR IEA-R1 PARA A
PESQUISA NUCLEAR

Conselho editorial

André Costa e Silva

Cecilia Consolo

Dijon de Moraes

Jarbas Vargas Nascimento

Luis Barbosa Cortez

Marco Aurélio Cremasco

Rogério Lerner

Blucher Open Access

PAULO SERGIO CARDOSO DA SILVA
GUILHERME SOARES ZAHN
FRANCISCO DE ASSIS SOUZA
(organizadores)

CONTRIBUIÇÕES DO
REATOR IEA-R1 PARA A
PESQUISA NUCLEAR
WARP2: II Workshop Anual do
Reator de Pesquisas

21 e 22 de novembro de 2019
Centro do Reator de Pesquisas
Instituto de Pesquisas Energéticas e Nucleares

2022

Contribuições do Reator IEA-R1 para a Pesquisa Nuclear

WARP2: II Workshop Anual do Reator de Pesquisas

© 2022 Paulo Sergio Cardoso da Silva, Guilherme Soares Zahn e Francisco de Assis Souza

Editora Edgard Blücher Ltda.

Publisher Edgard Blücher

Editor Eduardo Blücher

Coordenação editorial Jonatas Eliakim

Produção editorial Thaís Costa

Diagramação Taís do Lago

Capa Laércio Flenic

Blucher

Rua Pedroso Alvarenga, 1245, 4º andar
04531-934 – São Paulo – SP – Brasil
Tel 55 11 3078-5366
contato@blucher.com.br
www.blucher.com.br

Segundo Novo Acordo Ortográfico, conforme 5. ed.
do *Vocabulário Ortográfico da Língua Portuguesa*,
Academia Brasileira de Letras, março de 2009.

É proibida a reprodução total ou parcial por quaisquer
meios, sem autorização escrita da Editora.

Todos os direitos reservados pela Editora
Edgard Blücher Ltda.

Dados Internacionais de Catalogação na Publicação (CIP)

Angélica Ilacqua CRB-8/7057

Workshop anual do reator de pesquisas (2. : 2019 :
São Paulo)
Contribuições do reator IEA-R1 para a pesquisa
nuclear WARP 2 / organizado por Paulo Sergio Cardoso
da Silva, Guilherme Soares Zahn, Francisco de Assis
Souza. -- São Paulo : Blucher, 2022.
478 p : il.
21 e 22 de novembro de 2019 - Centro do Reator de
Pesquisas
Instituto de Pesquisas Energéticas e Nucleares
Bibliografia
ISBN 978-65-5550-147-6 (impresso)
ISBN 978-65-5550-148-3 (eletrônico)
1. Pesquisa nuclear 2. Física nuclear I. Título II. Silva,
Paulo Sergio Cardoso da III. Zahn, Guilherme Soares IV.
Souza, Francisco de Assis IV. IPEN

21-5617

CDD 539.7

Índices para catálogo sistemático:

1. Pesquisa nuclear

COMITÊ ORGANIZADOR

Paulo Sergio Cardoso da Silva

Guilherme Soares Zahn

Francisco de Assis Souza

COMITÊ CIENTÍFICO

Paulo Sergio Cardoso da Silva

Guilherme Soares Zahn

Francisco de Assis Souza

Frederico Antônio Genezini

APOIO

O Comitê Organizador agradece o apoio do Instituto de Pesquisas Energéticas e Nucleares (IPEN), do Departamento de Ensino do IPEN e da Marinha do Brasil, para a realização do II Workshop Anual do Reator de Pesquisas.



AMBIENTE DE ARQUIVOS PARA INTEGRAÇÃO DE LABORATÓRIO E EXTENSÃO WEB

*Franco Brancaccio, Mauro S. Dias, Marina F. Koskinas,
Denise S. Moreira e Fábio de Toledo*

Centro do Reator de Pesquisas – IPEN-CNEN/SP
Av. Professor Lineu Prestes, 2242
05508-000 São Paulo – SP
fbrancac@ipen.br

RESUMO

O Laboratório de Metrologia Nuclear (LMN) desenvolve sistemas de medida, metodologias e ferramentas de software como suporte às atividades principais. Entre os sistemas, podem-se citar os que utilizam métodos secundários para a determinação da atividade de amostras radioativas, utilizando câmaras de ionização, bem como os que utilizam métodos primários para a padronização de radionuclídeos, empregando arranjos de detecção compostos por contadores proporcionais, cintiladores e detetores de germânio hiperpuro (HPGe). A elaboração de software está usualmente associada à aquisição e análise de dados, para cálculos, ajuste de dados experimentais (*fitting*), ou a funções coadjuvantes, como a calibração de sistemas, simulação e previsão de resultados pelo método de Monte Carlo. Os programas até então desenvolvidos operam em modo local (*offline*) e a

configuração necessária à sua correta operação é efetuada por arquivos de texto (ASCII) pouco amigáveis, contendo parâmetros e opções (*flags*). A portabilidade de dados (transporte e utilização em diferentes máquinas e plataformas) constitui outra dificuldade inerente à abordagem *offline*. A proposta do presente trabalho consiste em desenvolver aplicações baseadas em Web, disponibilizando serviços e informações, respectivamente, em servidores de software e de Bancos de Dados (*Data Base – DB*). Um dos principais requisitos deste projeto consiste em estabelecer formatos padrão para os dados (esquemas) e uma linguagem amigável (*human friendly*) para sua representação (notação), compatível com ambiente Web/DB. A expressão “Ambiente de Arquivos” sintetiza a dupla definição, esquema/notação, tema deste trabalho.

1. INTRODUÇÃO

O Laboratório de Metrologia Nuclear (LMN), fundado em 1964, desenvolve diversas atividades, como:

- SERVIÇOS:
 - produção de amostras radioativas padronizadas;
 - ensino em Pós Graduação (Mestrado e Doutorado);
- METROLOGIA [1-6]:
 - mapeamento do fluxo neutrônico do reator IEA-R1;
 - padronização de radionuclídeos e amostras radioativas;
 - determinação de parâmetros nucleares:
 - seções de choque para nêutrons;
 - intensidades de transições gama;
 - meias-vidas de radionuclídeos e de transições metaestáveis.
- ATUALIZAÇÃO E DESENVOLVIMENTO [7-26]:
 - sistemas e metodologias de medida;
 - software (aquisição/análise de dados; calibração; simulação; ferramentas auxiliares).

O desenvolvimento de software, classe à qual pertence o presente trabalho, pode ser subdividido em categorias, de acordo com a finalidade:

- calibração de sistemas e obtenção de curvas de eficiência de detecção;
- aquisição de dados em sistemas de medida para obtenção de “informação bruta”;
- análise de dados de aquisição para obtenção de parâmetros, gráficos, tabelas etc.;
- ajuste de dados experimentais (*fitting*);
- simulação para obtenção de dados e previsão de resultados pelo método de Monte Carlo.

Como em outras atividades, o desenvolvimento de software busca por novas tendências e ferramentas. Em trabalho recentemente aceito para publicação (outubro de 2019) [26], descreve-se o programa de análise de coincidências por software, denominado *Coincidence Analysis Task* (CAT), primeiro aplicativo do LMN escrito em linguagem C++ a utilizar orientação a objetos, *Object-oriented Programming* (OOP). Técnicas OOP ampliam o conceito de reutilização de código, com a criação de classes e ferramentas para o encapsulamento de informações e funcionalidades, disponibilizadas em bibliotecas para uso em outros aplicativos. CAT inclui bibliotecas de uso geral, como:

- elementos e sistemas (descrição e configuração de sistemas, arranjos e componentes);
- elementos químicos e (rádio)isótopos (número atômico e de massa; constantes);
- grandezas físicas (prefixo, unidade, valor, incerteza), com sobrecarga de funções matemáticas para suporte à propagação de incertezas.
- pulsos elétricos (sinais de detecção das radiações: altura e instante de ocorrência);
- espectros de distribuição de energia (histogramas);
- distribuição de tempos de coincidências beta/gama;
- tabelas de dados de extrapolação de eficiência de contagem beta;
- ajuste linear de dados de extrapolação.

O programa CAT foi, assim, brevemente apresentado, por constituir a matéria escolhida para desenvolvimento do ambiente de arquivos. O projeto tem o objetivo de contemplar diferentes áreas e aplicações, tendo início com a reformulação de CAT, para a criação de uma nova versão Web, denominada Wild CAT (WCAT).

2. AMBIENTE DE ARQUIVOS

O software desenvolvido no LMN segue basicamente o paradigma procedimental, que utiliza funções e rotinas, em código estruturado. Como mencionado, CAT introduz conceitos de OOP, incluindo bibliotecas que podem ser “ligadas” (*linking*) a qualquer programa em tempo de compilação, de modo a disponibilizar toda funcionalidade incorporada (algoritmos e funções; classes e objetos; etc.), sem a necessidade de reescrever ou incluir código fonte (reutilização de código objeto, previamente compilado).

Independentemente da abordagem adotada, os programas operam em modo *offline* e os parâmetros necessários devem ser editados manualmente pelo operador/ usuário. Os arquivos de entrada e saída (dados, configurações e resultados) possuem formato texto (ASCII) sem padrão definido (*custom*), dificultando o intercâmbio de dados entre programas e a sua utilização (dados e programas) em máquinas ou plataformas diferentes.

Para contornar os problemas apontados é necessário adotar estratégias como a padronização de formatos para dados e arquivos (compatível com diferentes linguagens e plataformas), a utilização de Bancos de Dados (*Data Base – DB*) e o uso de abordagem de desenvolvimento baseada em Web.

Para representação de dados escolheu-se o padrão JSON (*JavaScript Object Notation*) [27], por atender aos requisitos mencionados:

- suportado pelas principais linguagens de programação, como JavaScript (nativo), Node.js (idem), C++ e Python [28-31], entre muitas outras;
- amplamente utilizado em ambiente Web como formato de intercâmbio de mensagens e dados;
- sua versão em notação binária, BSON [32] (*Binary JSON*), é utilizada por gerenciadores DB como MongoDB [33], ferramenta também adotada neste projeto.

O formato JSON baseia-se na estrutura {“chave”: “valor”}, como mostra a Figura 1, na qual se define um objeto “range”, representando um intervalo de valores de tensão elétrica (V), em ponto flutuante. Incluem-se uma faixa principal, [0.0, 10.0], uma região de sobre escala, (10.0, 13.0), além de informações auxiliares: cabeçalho, conjunto de autores [*array*] e versão.

Figura 1 – Versão inicial de objeto “range” em notação JSON, idealizado para o programa WCAT.

```
{
  "description": "Definition of a numeric range",
  "authors": [
    {"id": "authorDB#Denise", "name": "Denise Simões Moreira"},
    {"id": "authorDB#Fabio", "name": "Fábio de Toledo"},
    {"id": "authorDB#Franco", "name": "Franco Brancaccio"},
    {"id": "authorDB#Marina", "name": "Marina Fallone Koskinas"},
    {"id": "authorDB#Mauro", "name": "Mauro da Silva Dias"}
  ],
  "version": {"startYear": 2017, "versionYear": 2019, "number": "1.0.0"},
  "range": {}
}
```

A complexidade dos objetos em formato JSON varia conforme a necessidade. Como exemplo, apresentam-se três versões simples de um objeto “pessoa”, contendo apenas o nome completo como informação:

- “pessoa”: {“nome”: “Joaquim Leme”};
- “pessoa”: {“primeiroNome”: “Joaquim”, “sobrenome”: “Leme”};
- “pessoa”: {“nome”: {“primeiro”: “Joaquim”, “família”: “Leme”}}.

Deste modo, o objeto “pessoa” poderia incluir outros atributos, como peso, estatura, data e local de nascimento, endereço, e-mail, formação, profissão, local de trabalho, interesses etc.

2.1. Informações e Representação JSON

Toda informação em WCAT será representada em notação JSON, para armazenamento em bancos de dados MongoDB. Enumeram-se abaixo algumas categorias de objetos JSON idealizados para WCAT, com exemplos ilustrativos quando possível:

- variáveis: “count”: {“value”: 27479};
- objetos: “pulse”: {“height”: [7.13, “V”], “timeStamp”: [12345678.9, “ns”]}.
- arquivos (DB):

- elementos e itens (descrição de sistemas de medida e de análise: detetores, módulos eletrônicos e software; amostras e fontes radioativas etc.);
- constantes físicas e parâmetros (meias-vidas de decaimento; intensidades de emissão gama; seções de choque etc.);
- configuração de aquisição (data; descrição; id da amostra alvo; tempo de medida; id do operador; nomes dos arquivos de dados etc.);
- dados de aquisição (arquivos de dados: valores obtidos para análise posterior), compostos por registros dos pulsos de detecção nuclear: [amplitude, instante], como no exemplo acima (objeto “pulse”);
- configuração de análise (id da medida; data; data de referência; parâmetros; id do operador; nomes dos arquivos de resultados etc.);
- resultados de análise (espectros: beta, gama e distribuição de tempos de coincidências; tabela de extrapolação; dados de ajuste; relatório síntese etc.).

2.2. Desenvolvimento de Software

A abordagem de desenvolvimento Web também permite efetuar a hospedagem de software, oferecendo acesso a programas ou serviços. Utilizando navegadores (*browsers*) ou aplicativos apropriados, em diferentes plataformas ou dispositivos (*desk* ou *laptops*, *tablets*, *smartphones*), é possível acessar as informações armazenadas e executar qualquer tarefa disponível, como *fitting* de dados experimentais, elaboração de gráficos, análise de conjuntos de dados, entre outros.

A padronização (ambiente de arquivos) também é requisito para que diferentes serviços tenham acesso compartilhado a dados (ver exemplo no final desta seção).

Algumas vantagens das técnicas de desenvolvimento Web são apresentadas na Tabela 1, em que se efetua, de forma simplificada, o confronto entre duas abordagens, conforme as seguintes considerações:

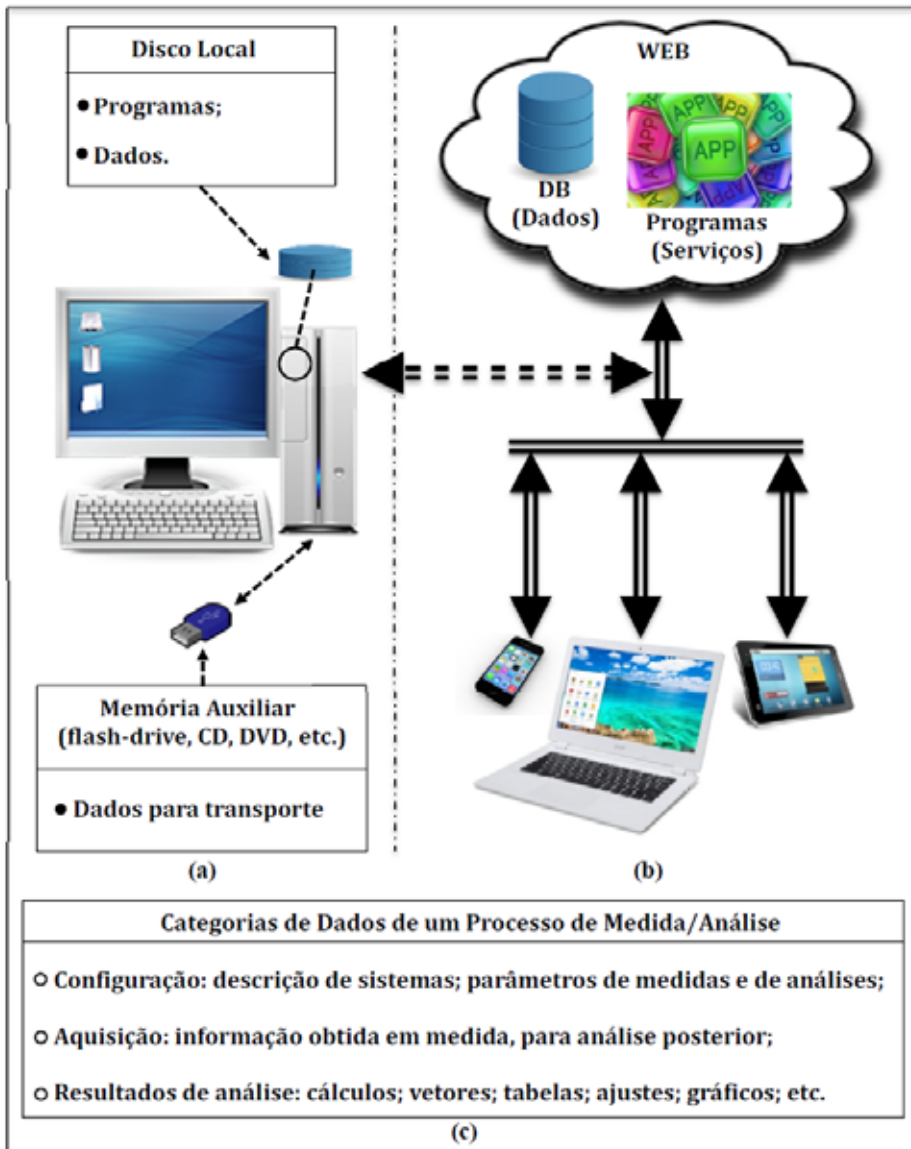
- a abordagem de programação adotada até o presente (conforme descrição no início da seção 2) recebe a denominação “Convencional”;
- à proposta de dados e serviços globalmente acessíveis, denomina-se “Web”;
- estabelece-se um paralelo entre as abordagens (dispostas em duas colunas), situando-se horizontalmente os critérios de comparação (linhas).

Tabela 1 – Comparação entre abordagens de desenvolvimento de software, aqui denominadas Convencional e Web

Convencional	Web
Instalação de Programas: Necessária em todas as máquinas em que se deseja executá-los.	Instalação local de Programas: <i>Browser</i> ou aplicativo de acesso Web (ver Hospedagem em Servidores).
Plataforma: Normalmente monoplataforma (Windows, Linux ou Mac); É necessário (re)compilar e instalar o programa em cada plataforma.	Plataforma: Multiplataforma (todas: Windows, Linux, Mac e móveis); Compilação única (<i>Deploy</i> em Servidor).
Operação em Modo Offline: Dados e parâmetros acessíveis apenas localmente, por meio de discos, memórias flash etc.; Programas: v. Instalação de Programas .	Hospedagem em Servidores: Dados e parâmetros acessíveis globalmente, em bancos de dados (DB); Programas acessíveis em browsers ou aplicativos com acesso Web.
Execução: Poucos programas executam todas as tarefas, às vezes, apenas um programa generalista (configuração monolítica).	Execução: Serviços executam tarefas especializadas, como: leitura de dados, análise e cálculo; ajuste de dados (<i>fitting</i>); gráficos; edição de dados em DB etc.
Manutenção e Atualização: Trabalhosa, exigindo recompilação e reinstalação completas em todas as máquinas.	Manutenção e Atualização: Somente o serviço alterado deve ser reinstalado no respectivo servidor.

A Figura 2 esquematiza as diferenças entre as abordagens de programação apresentadas na Tabela 1 e sintetiza as diferentes categorias de dados de um processo de medida e análise.

Figura 2 – Representação das abordagens de programação (a) e (b) e categorização dos dados de medida e análise (c).



(a) Convencional, com acesso local (restrito) a dados e programas;

(b) Web, onde dados e serviços são acessíveis globalmente;

(c) Tipos de dados de um processo (descritivos; parâmetros; resultados).

Serviços hospedados em Web (Tabela 1 e Figura 2) são normalmente independentes, podendo-se utilizar diferentes linguagens em sua codificação (Java, JavaScript, Python, C++ etc.). Entretanto, diferentes serviços podem estabelecer

comunicação para intercâmbio de dados que compartilham o mesmo formato (esquema). Como exemplo, considere-se o seguinte cenário:

- **Data:** conjunto de dados, em formato padrão, obtidos em uma medida;
- **Result:** esquema padrão para dados, idealizado para a produção de gráficos;
- **MyCalc:** serviço de processamento:
 - Entrada: objeto **Data**;
 - Saída: objeto **CalcResult** (resultado, com esquema **Result**);
- **MyFit:** serviço para ajuste de dados experimentais (*fitting*):
 - Entrada: objeto com esquema **Result** (obtidos com **MyCalc** ou outros);
 - Saída: objeto **FitResult** (resultado, com esquema **Result**);
- **MyGraph:** serviço que produz gráficos:
 - Entrada: objetos com esquema **Result** (obtidos com **MyCalc**, **MyFit**, ou outros);
 - Saída: objeto de imagem, **GraphResult** (resultado: gráfico desejado);

Neste exemplo, os objetos **CalcResult** (saída de **MyCalc**) e **FitResult** (saída de **MyFit**) compartilham o esquema **Result**, também utilizado para os dados de entrada do serviço **MyGraph**, de modo que este último pode gerar curvas para os dois resultados (processo e ajuste). O esquema **Result** encapsula a informação necessária à obtenção de gráficos: dados; título; eixos (nomes; tipo: linear, logarítmico); categoria (dispersão; linhas; barras; histograma; pizza; área; 2D, 3D); etc. O serviço **MyGraph** pode ser utilizado por qualquer outro que entregue objetos com esquema **Result**. Em particular, observa-se que o mesmo esquema pode ser utilizado para funções distintas (neste caso, *fitting* e gráfico: **MyFit** utiliza o esquema **Result** para seus objetos de entrada e de saída).

3. CONCLUSÃO

Este trabalho apresenta os requisitos para desenvolvimento de aplicações Web, bem como as medidas adotadas pelo LMN para atingir esse objetivo. Destaca-se a necessidade de padronização dos formatos da informação, para permitir o intercâmbio de dados entre serviços e o acesso a bancos de dados (DB), estabelecendo-se o formato JSON como língua franca. Como projeto

piloto, optou-se por converter o programa CAT para análise de coincidências por software em uma nova versão Web, denominada Wild CAT (WCAT). Arquivos e variáveis internas utilizados por CAT estão sendo alterados para o formato JSON (dados de entrada e saída; configurações; etc.). Testes de software estão em curso (programação Web; MongoDB; *inter-process communication*, IPC; processos *child*; produção de gráficos; *big numbers* para compressão de dados em base 64; *data fitting*; etc.). Funcionalidades previstas para inclusão em CAT serão desenvolvidas como serviços em WCAT, como busca de picos, compensação de contagem de fundo (linha de base), gráficos e ajuste de dados experimentais a qualquer função matemática (CAT dispõe somente de ajuste à função de primeiro grau). Os serviços desenvolvidos em WCAT ficam disponíveis a outros projetos, citando-se como exemplo a aplicação Web prevista para Análise de Ativação Neutrônica, que poderá utilizar os serviços de busca de picos e cálculo de áreas, para a determinação da composição das amostras alvo e a quantificação de elementos de interesse.

REFERÊNCIAS

1. KOSKINAS, M. F.; DIAS, M. S. Measurement of the gamma-ray probability per decay of ^{31}Si . *Applied Radiation and Isotopes*, v. 44, p. 1209-1211, 1993.
2. DIAS, M. S.; KOSKINAS, M. F. Activity determination of ^{75}Se radioactive solution by absolute method. *Phys. Med. Biol.*, v. 39a, p. 798, 1994.
3. ATTIE, M. R. P.; KOSKINAS, M. F.; DIAS, M. S.; FONSECA, K. A. Absolute disintegration rate measurements of ^{67}Ga . *Applied Radiation and Isotopes*, Elsevier, v. 49, n. 9-11, p. 1175-1177, 1998.
4. KOSKINAS, M. F.; FONSECA, K. A.; DIAS, M. S. Disintegration rate measurement of a ^{152}Eu solution. *Applied Radiation and Isotopes*, Elsevier, v. 56, p. 441-445, 2002.
5. BACCARELLI, A. M.; DIAS, M. S.; KOSKINAS, M. F.; BRANCACCIO, F. Standardization of ^{18}F by means of $4\pi(\text{PS})\beta\text{-}\gamma$ plastic scintillator coincidence system. *IEEE Transactions on Nuclear Science*, v.55, n. 3, p. 1767-1773, 2008.
6. DIAS, M. S.; BRANCACCIO, F.; TOLEDO, F.; KOSKINAS, M. F. Disintegration rate, gamma-ray emission probabilities and metastable half-life measurements of ^{67}Ga . *Applied Radiation and Isotopes*, Elsevier, ICRM 2013, v. 87, p. 126-131, 2014.
7. DIAS, M. S. *Calibração de um sistema de câmara de ionização de poço $4\pi\text{-}\gamma$ para medidas de atividade de radionuclídeos*. 1978. Dissertação (Mestrado) –

Instituto de Energia Atômica, Escola Politécnica, Universidade de São Paulo (USP), São Paulo, 1978.

8. DIAS, M. S.; RENNER, C. Si(Li) efficiency curve for X-ray parallel beam. *Nuclear Instruments and Methods*, v. 193, p. 91, 1982.

9. KOSKINAS, M. F. *Desenvolvimento de um sistema de coincidência para a medida absoluta da atividade de radionuclídeos empregando detectores de barreira de superfície*. 1988. 140 p. Tese (Doutorado) – Instituto de Pesquisas Energéticas e Nucleares (IPEN), Universidade de São Paulo (USP), 1988.

10. KOSKINAS, M. F.; DIAS, M. S. Coincidence system for radionuclide standardization using surface barrier detectors. *Nuclear Instruments and Methods in Physics Research*, v. A280, p. 327-331, 1989.

11. DIAS, M. S.; KOSKINAS, M. F. Accidental summing correction in ^{125}I activity determination by the sum-peak method. *Applied Radiation and Isotopes*, v. 9, p. 945- 948, 1995.

12. BRANCACCIO, F. *Automatização de um sistema de medidas de correntes produzidas por câmaras de ionização e aplicação na calibração do ^{18}F e ^{153}Sm* . 2002. 94 p. Dissertação (Mestrado em Tecnologia Nuclear – Aplicações) – Instituto de Pesquisas Energéticas e Nucleares (IPEN), Universidade de São Paulo (USP), São Paulo, 2002.

13. BRANCACCIO, F.; DIAS, M. S.; KOSKINAS, M. F. Automatic system for ionization chamber current measurements. *International Journal of Applied Radiation and Isotopes*, Elsevier, v. 61, p. 1339-1342, 2004.

14. DIAS, M. S.; PIUVEZAM FILHO, H.; BACCARELLI, A. M.; TAKEDA, M. N.; KOSKINAS, M. F. Monte Carlo simulation of β - γ coincidence system using plastic scintillators in 4π geometry. *Nuclear Instruments and Methods in Physics Research*, Elsevier, v. A 580, p. 380-383, 2007.

15. DIAS, M. S.; TAKEDA, M. N.; KOSKINAS, M. F. Application of Monte Carlo simulation to the prediction of extrapolation curves in the coincidence technique. *Applied Radiation and Isotopes*, Elsevier, v. 64, p. 1186-1192, 2006.

16. TOLEDO, F.; BRANCACCIO, F.; LAPOLLI, A.; DIAS, M. S. Design of electronic system with simultaneous registering of pulse and event time applied to $4\pi\beta$ - γ coincidence method. In: IEEE 2008 NUCLEAR SCIENCE SYMPOSIUM; MEDICAL IMAGING CONFERENCE; 16TH ROOM TEMPERATURE SEMICONDUCTOR DETECTOR WORKSHOP, 19-25 out. 2008, Dresden/Alemanha. *2008 IEEE Nuclear Science Symposium Conference*

Record, 2008. p. 1003-1005. DOI: 10.1109/NSSMIC.2008.4774564. Disponível em: <https://ieeexplore.ieee.org/document/4774564>.

17. DIAS, M. S. *LOGFIT*: código para ajustes logarítmicos. Relatório interno, LMN IPEN, 1997.

18. DIAS, M. S. *LINFIT*: código para ajustes lineares. Relatório interno, LMN IPEN, 1999.

19. DIAS, M. S. *ALPINO*: código para obtenção de atividades e eficiências de contagem com espectrômetro REGe. Relatório interno, LMN IPEN, 2000.

20. DIAS, M. S. *DILUIP7*: código para cálculo de atividade, com o sistema de coincidências, convencional com contadores. Relatório interno, LMN IPEN, 2000).

21. DIAS, M. S. *CONTAC9*: código para cálculo de atividades do sistema de coincidências TAC. Relatório interno, LMN IPEN, 2001.

22. DIAS, M. S.; TAKEDA, M. N.; KOSKINAS, M. F. Cascade summing corrections for REGe spectrometers by the Monte Carlo method. *Applied Radiation and Isotopes*, v. 56, p. 105- 109, 2002.

23. DIAS, M. S. *SCTAC*: código para cálculo de atividade baseado em medidas de contagem de coincidências por *Software*. Relatório interno, IPEN-CNEN/SP, 2010.

24. DIAS, M. S.; TAKEDA, M. N.; TOLEDO, F.; BRANCACCIO, F.; TONGU, M. L. O.; KOSKINAS, M. F. Improvements in the Monte Carlo code for simulating $4\pi \beta(\text{PC})-\gamma$ coincidence system measurements. *Nuclear Instruments and Methods in Physics Research*, Elsevier, NIMA-D-12-00550R1, 2013.

25. BRANCACCIO, F. *Metodologia de aquisição de dados e análise por software, para sistemas de coincidências $4\pi\beta-\gamma$ e sua aplicação na padronização de radionuclídeos, com ênfase em transições metaestáveis*. 2013. 146 p. Tese (Doutorado em Tecnologia Nuclear – Aplicações) – Instituto de Pesquisas Energéticas e Nucleares (IPEN), Universidade de São Paulo (USP), São Paulo, 2013. DOI: 10.11606/T.85.2013.tde-03102013-104013. Disponível em: <https://www.teses.usp.br/teses/disponiveis/85/85131/tde-03102013-104013/pt-br.php>.

26. BRANCACCIO, F.; DIAS, M. S.; KOSKINAS, M. F.; MOREIRA, D. S.; TOLEDO, F. Data analysis software package for radionuclide standardization with a digital coincidence counting system. *Applied Radiation and Isotopes*, v. 155, p. 1-10, 2020. Disponível em: <https://doi.org/10.1016/j.apradiso.2019.108900>.

27. JSON. Site oficial. Disponível em: <https://www.json.org>.

28. JavaScript. Site oficial. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.
29. Node.js. Site oficial. Disponível em: <https://nodejs.org/en>.
30. C++. Site oficial. Disponível em: <http://www.stroustrup.com/C++.html>.
31. Python. Site oficial. Disponível em: <https://www.python.org>.
32. BSON. Site oficial. Disponível em: <http://bsonspec.org>.
33. MongoDB. Site oficial. Disponível em: <https://www.mongodb.com>.

