# Resumo

Neste artigo apresentamos o Modelo de Dados para Organização Representação da Informação Museológica (MDORIM) do Elucidário.art, app em desenvolvimento que tem como objetivo ser uma ferramenta de gestão e divulgação de coleções museológicas. O MDORIM utiliza o *Linked-art* para a representação de informações e a norma *Standard Procedures for Collections Recording Used in Museums* (SPECTRUM) na versão 5.1. O ciclo de vida de um objeto de museu pode descrever a sua trajetória desde antes de sua aquisição até a saída do acervo, podendo passar por etapas de conservação, restauração, exposição, pesquisa, empréstimos, entre outros. O SPECTRUM é uma norma que descreve 21 procedimentos para gestão de coleções museológicas, sendo 9 deles obrigatórios para acreditação de um museu à *Collections Trust*, organização britânica que define padrões de gestão museológica. O SPECTRUM também é recomendado pelo ICOM e utilizado por museus do mundo todo, inclusive no Brasil. O *Linked-Art* é um modelo de dados para aplicações baseado no CIDOC-CRM. Focado em representação de dados para museus de arte, o *Linked-Art* foi criado por um grupo de trabalho internacional do CIDOC e representantes de instituições museológicas. O MDORIM estabelece uma ponte entre o *Linked-Art* e o *SPECTRUM*, o que permite a gestão da documentação de forma interoperável utilizando padrões comuns entre museus. Para isso, analisaremos as documentações do SPECTRUM e do *Linked-art* para criar um mapeamento entre os dois. O MDORIM também introduz classes de informação para auditoria e controle como histórico de edições e níveis de permissão por tipo de usuário.

O ciclo de vida de um objeto museológico pode descrever a sua trajetória desde antes de sua aquisição até a saída do acervo, podendo passar por etapas de conservação, restauração, exposição, pesquisa, empréstimos, entre outros. O Modelo de Dados para Organização e Representação da Informação Museológica (MDORIM) do Elucidário.art utiliza o *Linked-art* para representação destas informações e implementa a norma *Standard Procedures for Collections Recording Used in Museums* (SPECTRUM) na versão 5.1 que descreve 21 procedimentos para gestão de coleções museológicas, sendo 9 deles principais e obrigatórios para acreditação de um museu à *Collections Trust*, organização britânica que define padrões para gestão museológica. O SPECTRUM também é recomendado pelo *International Council of Museums* (ICOM) e utilizado por museus de todo o mundo, inclusive no Brasil. O *Linked-Art* é um modelo de dados para aplicações baseado no CIDOC *Conceptual Reference Model* e é focado em representação de dados para museus de arte. O *Linked-Art* foi criado por um grupo de trabalho internacional do CIDOC e representantes de instituições museológicas. O MDORIM busca estabelecer uma ponte entre o Linked-Art e o SPECTRUM, permitindo a gestão da documentação de maneira interoperável utilizando um padrão comum entre os museus. Para isso, analisaremos as documentações do SPECTRUM e do *Linked-art* e criaremos um mapeamento entre os dois, além disso, o MDORIM também introduz classes de informação para auditoria e controle como histórico de edições e níveis de permissão por tipo de usuário.

Neste artigo apresentaremos o Modelo de Dados para Organização Representação da Informação Museológica (MDORIM) do Elucidário.art, app em desenvolvimento que tem como objetivo ser uma ferramenta de gestão e divulgação de coleções museológicas. O MDORIM utiliza o *Linked-art* para a representação das informações e a norma *Standard Procedures for Collections Recording Used in Museums* (SPECTRUM) na versão 5.1. O ciclo de vida de um objeto de museu pode descrever a sua trajetória desde antes de sua aquisição até a saída do acervo, podendo passar por etapas de conservação, restauração, exposição, pesquisa, empréstimos, entre outros. O SPECTRUM é uma norma que descreve 21 procedimentos para gestão de coleções museológicas, sendo 9 deles principais e obrigatórios para acreditação de um museu à *Collections Trust*, organização britânica que define padrões para gestão museológica. O SPECTRUM também é recomendado pelo ICOM e utilizado por museus de todo o mundo, inclusive no Brasil. O *Linked-Art* é um modelo de dados para aplicações baseado no CIDOC-CRM. Focado em representação de dados para museus de arte, o *Linked-Art* foi criado por um grupo de trabalho internacional do CIDOC e representantes de museus. O MDORIM busca estabelecer uma ponte entre o *Linked-Art* e o *SPECTRUM*, permitindo a gestão da documentação de maneira interoperável utilizando padrões comuns entre os museus. Para isso, analisaremos as documentações do SPECTRUM e do *Linked-art* e criaremos um mapeamento entre os dois, além disso, o MDORIM também introduz classes de informação para auditoria e controle como histórico de edições e níveis de permissão por tipo de usuário.

# Introdução

# Metodologia

Para a realização deste trabalho, iremos analisar os procedimentos do SPECTRUM e suas Unidades de Informação, a documentação do Linked-art e buscaremos artigos relacionados aos dois padrões em bases de dados internacionais.

# Standard ProCedures for Collections Recording Used in Museums (Spectrum)

O Spectrum é um padrão para gestão de coleções museológicas criado no Reino Unido por Alice Grant e publicada inicialmente em 1994 pela *Museum Documentation Association* (MDA), em 2005 passa a ser editada por Gordon McKenna (3ª Edição) (MATOS, 2014). O Spectrum é utilizado por todos os museus acreditados pela *Collections Trust* (COLLECTIONS TRUST, 2022) no Reino Unido e diversos outros ao redor do mundo, e define 21 procedimentos para gestão de coleções museológicas e unidades de informações utilizadas nos procedimentos.

Em 2014, com um esforço conjunto da Secretaria de Estado de Cultura, Associação de Amigos do Museu do Café e Pinacoteca do Estado de São Paulo, a versão 4.0 do Spectrum foi traduzida para o português-brasileiro [[1]](#footnote-23), baseado na versão previamente traduzida para o português pela Universidade do Porto em Portugal. Nesta versão, originalmente editada em 2011 por Alex Dawson e Susie Hillhouse, tem sem nome alterado de “UK Museum Documentation Standard” para “UK Collections Management Standard”, e apresenta uma nova estrutura que passa a publicar os procedimentos e unidades de informação em documentos separados, e os procedimentos passam a ser apresentados em fluxos de trabalho (COLLECTIONS TRUST, [s.d.]).

A versão 5.0 é publicada em 2017 e foi editada por Kevin Gosling e Gordon McKenna. Esta versão passa a apresentar os procedimentos em fluxos de trabalho e em texto utilizado em versão anteriores. O padrão também foi reescrito de uma forma mais clara e concisa, criando uma distinção entre “padrão” e “orientação”:

* **Padrão**: é o que deve ser feito, é obrigatório para a acreditação à *Collections Trust*;
* **Orientação**: é o que pode ser feito, é opcional e são apenas sugestões de boas práticas.

A versão 5.0 também substitui o procedimento “Documentação retrospectiva” por dois novos procedimentos: “Planejamento da documentação” e “Inventário”, para melhorar a distinção no que é válido como documentação retrospectiva (COLLECTIONS TRUST, [s.d.]).

A versão 5.1, publicada em 2022 como parte da campanha “rethink cataloguing” (COLLECTIONS TRUST, [s.d.]), modifica os procedimentos de Catalogação e Uso das coleções para estimular uma abordagem mais inclusiva nestas atividades (COLLECTIONS TRUST, [s.d.]).

Utilizaremos a versão 5.1 em nossos estudos por ser a versão mais recente do Spectrum.

## Procedimentos

O Spectrum define 21 procedimentos para gestão de coleções museológicas (COLLECTIONS TRUST, 2023), são os procedimentos que um museu irá utilizar em todo o ciclo de vida de um objeto musealizado, desde a sua aquisição até a sua desincorporação.

Um procedimento Spectrum é descrito como um fluxo de trabalho, composto por uma definição, notas de escopo, o padrão Spectrum (o que o museu deve alcançar) e procedimento sugerido. O padrão Spectrum que deve ser alcançado é dividido em duas partes: políticas e requisitos mínimos. As políticas são questões que o museu deve considerar ao definir o seu procedimento, e os requisitos mínimos são as atividades que devem ser realizadas para que o procedimento seja considerado minimamente completo. O procedimento sugerido também é apresentado em duas partes: um diagrama de fluxo de trabalho e uma descrição textual do procedimento (COLLECTIONS TRUST, 2022).

Dos 21 procedimentos, 9 são obrigatórios para acreditação à *Collections Trust* e são os que o museu irá utilizar em todos os objetos musealizados, os outros 12 são opcionais e o museu pode escolher quais utilizar de acordo com a sua necessidade. Os procedimentos podem ser entendidos como uma cadeia de atividades que se interconectam, em que um procedimento pode desencadear o outro.

Os procedimentos obrigatórios são:

1. Entrada de Objetos

* Procedimento para registrar a entrada de objetos no museu, seja por doação, compra, empréstimo, etc.

1. Aquisição e adesão

* Procedimento para registrar a aquisição de objetos, seja por compra, doação, etc.

1. Localização e controle de movimentação

* Procedimento para registrar a localização de objetos no museu e o controle de movimentação dos objetos.

1. Inventário

* Procedimento para registrar o inventário dos objetos musealizados.

1. Catalogação

* Procedimento para registrar a catalogação dos objetos musealizados.

1. Saída de objetos

* Procedimento para registrar a saída de objetos do museu, seja por empréstimo, doação, venda, etc.

1. Entrada de empréstimos

* Procedimento para registrar a entrada de objetos emprestados ao museu.

1. Saída de empréstimos

* Procedimento para registrar a saída de objetos emprestados pelo museu.

1. Planejamento da documentação

* Procedimento para registrar o planejamento da documentação dos objetos musealizados.

Os procedimentos não obrigatórios são:

1. Uso das coleções

* Procedimento para registrar como coleções e dados associados são utilizados pelo museu e por outros usuários.

1. Verificação de condições e avaliação técnica

* Procedimento para registrar a verificação de condições e avaliação técnica dos objetos musealizados.

1. Cuidados e conservação de coleções

* Procedimento para registrar os cuidados e conservação dos objetos musealizados.

1. Valoração

* Procedimento para registrar a valoração dos objetos musealizados.

1. Seguro e indenização

* Procedimento para registrar o seguro e indenização dos objetos musealizados.

1. Plano de emergência para coleções

* Procedimento para registrar o plano de emergência para coleções do museu.

1. Danos e perdas

* Procedimento para registrar os danos e perdas dos objetos musealizados.

1. Desacessão e alienação

* Procedimento para registrar a desacessão e alienação dos objetos musealizados.

1. Gestão de direitos

* Procedimento para registrar a gestão de direitos dos objetos musealizados.

1. Reprodução

* Procedimento para registrar a reprodução dos objetos musealizados.

1. Revisão da coleção

* Procedimento para registrar a revisão da coleção do museu.

1. Auditoria

* Procedimento para registrar a auditoria da coleção do museu.

## Unidades de informação

Além dos procedimentos, o Spectrum apresenta um conjunto de unidades de informação, ou metadados, para descrever os objetos, procedimentos e atividades realizadas pelo museu. As unidades de informação são divididas em três grupos que contêm metadados relacionados: grupo de informações sobre objeto (Object information groups), grupo de informações sobre procedimento (Procedure information groups) e grupo de informações sobre gestão (Record management information groups) (COLLECTIONS TRUST, 2017).

As unidades de informação são apresentadas com uma página para cada metadado contendo a definição, como registrar, exemplos, uso e a qual grupo pertence. Embora algumas unidades de informação oferecem uma hierarquia para atribuir maior contexto, como no caso do *Object name*, as informação podem ser facilmente representadas de forma tabular, como no exemplo a seguir.

Hierarquia das unidades de informações “Object name” (COLLECTIONS TRUST, 2017):

* Object name
  + Object name currency
  + Object name level
  + Object name note
  + Object name system
  + Object name type
  + Object name/title language

**Tabela 1: Exemplo de tabulação do metadado do Spectrum *Object name***

| object\_name | object\_name\_currency | object\_name\_note | object\_name\_language |
| --- | --- | --- | --- |
| Réquiem | current | nome original | pt-BR |

**Fonte**: Elaborado pelo autor. Exemplifica a tabulação do campo *object\_name* do Spectrum com o registro da gravura Réquiem de Maria Bonomi da Coleção Ema Klabin.

# Linked Art

Linked Art é uma comunidade formada por representantes de instituições ao redor do mundo como The Canadian Heritage Information Network (CHIN), J. Paul Getty Trust, The Frick Collection, Europeana, Louvre, Rijksmuseum, The Victoria and Albert Museum, entre outras, com o objeto de criar um modelo compartilhado baseado em *Linked Open Data* (LOD) para descrever Arte (LINKED ART, 2021a, 2021b). O projeto é coordenado por um quadro editorial em que Robert Sanderson (Yale University) e Emmanuelle Delmas-Glass (Yale Center for British Art) compartilham o posto de co-presidentes.

A comunidade Linked Art parte do conceito da usabilidade para a audiência correta, em que o maior público interessado em “dados” seriam os desenvolvedores que poderiam criar interfaces para o público final, desdobrando esta ideia em cinco princípios de design, temos:

1. **Abstração para a audiência correta**: desenvolvedores não precisam do mesmo acesso aos dados como os ontologistas;
2. **Poucas barreiras de entrada**: deve ser fácil começar a trabalhar com dados e construir algo: *“If it takes a long time to understand the model, ontology, sparql query syntax and so forth, then developers will look for easier targets”* (SANDERSON, 2018);
3. **Compreensível pela introspecção**: os dados devem ser compreensíveis pela leitura humana, usar JSON-LD é utilizar a linguagem que o desenvolvedor já compreende;
4. **Documentação com exemplos funcionais**: você nunca poderá intuir todas as regras dos dados, documentação serve para mapear os padrões que o desenvolvedor pode encontrar; e
5. **Poucas exceções em vez de vários padrões consistentes**: cada exceção em uma API é outra regra que o desenvolvedor precisa apreender: *“every exception is jarring, and requires additional code to manage. While not everything is homogenous, a set of patterns that manage exceptions well is better than many custom fields.”* (SANDERSON, 2018).

O modelo Linked Art tem como foco principal a descrição de recursos do patrimônio cultural artístico e atividades de museus. Para isto, o modelo utiliza um perfil do CIDOC *Conceptual Reference Model* (CIDOC-CRM) que combina a facilidade de uso do JSON-LD com a excelência do CRM, mas reduzindo sua complexidade para um conjunto de classes e propriedades que são mais comuns em 90% dos casos de usos de 90% das organizações (LINKED ART, 2021)

O Linked Art está sob desenvolvimento ativo e a versão atual é a 0.8.0 e é considerada instável, passível de mudanças. A versão 1.0.0 estava prevista para o final de 2021, mas foi atrasada por conta da pandemia de COVID-19 (LINKED ART, 2021).

O modelo apresenta 11 classes, ou entidades, sendo elas (LINKED ART, 2021):

1. **Concept** - tipos, materiais, idiomas, entre outros que sejam registros completos, ao contrário de referências externas;
2. **Digital Object** - imagens, vídeos, áudios, documentos, ou outros recursos digitais;
3. **Event** - eventos e atividades não específicas que estão relacionadas, mas não são parte de outra entidade;
4. **Groups** - grupos de pessoas, organizações, ou outras entidades;
5. **People** - pessoas;
6. **Physical Object** - objetos físicos, incluindo obras de arte, artefatos, edifícios, partes de objetos, entre outros;
7. **Place** - locais;
8. **Provenance Activity** - atividades de proveniência;
9. **Sets** - conjuntos de entidades;
10. **Textual Work** - obras textuais que merecem descrição como entidades únicas, como conteúdo de livro ou artigos, entre outros; e
11. **Visual Work** - conteúdo imagético que merece descrição como entidades únicas, como a imagem exibida em uma pintura ou desenho, entre outros.

## JSON para conectar dados: JSON-LD

*JavaScript Object Notation* (JSON) é um formato aberto de arquivo para intercâmbio de informações (W3C JSON-LD WORKING GROUP, 2014). Consiste em arquivos de fácil leitura por humanos e máquinas com a extensão .json. Mesmo que tenha se originado na sintaxe de objetos JavaScript, por isso seu nome, pode ser utilizado por diversos ambientes e linguagens de programação diferentes devido sua sintaxe simples (MDN WEB DOCS, 2023).

Um arquivo JSON pode ser expressado da seguinte forma:

**Exemplo 1**: JSON simples descrevendo uma obra de arte.

{  
 "title": "One and three chairs",  
 "author": "Joseph Kosuth"  
}

Mas o que este JSON representa para alguém não conhece o contexto artístico? O que é *“title”* e *“author”* para uma máquina, e quais valores eles podem receber?

*JSON for Linked Data* (JSON-LD) busca solucionar este problema adicionando uma camada de contexto ao JSON, ao fazer um link com a definição do vocabulário utilizado (SPORNY, 2012). Além disso é possível adicionar um identificador ao objeto, desta forma, o mesmo arquivo, porém agora com a definição do vocabulário e sua estrutura utilizando o contexto do Linked Art, pode ser expresso da seguinte forma:

**Exemplo 2**: JSON-LD descrevendo uma obra de arte.

{  
 "@context": "https://linked.art/ns/v1/linked-art.json",  
 "id": "https://www.moma.org/collection/works/81435",  
 "\_label": "One and three chairs",  
 "type": "HumanMadeObject",  
 "identified\_by": [  
 {  
 "type": "Identifier",  
 "\_label": "One and three chairs",  
 "classified\_as": [  
 {  
 "id": "http://vocab.getty.edu/aat/300404670",  
 "type": "Type",  
 "\_label": "Primary Name"  
 }  
 ],  
 "content": "One and three chairs"  
 }  
 ],  
 "produced\_by": {  
 "type": "Production",  
 "carried\_out\_by": [  
 {  
 "type": "Person",  
 "id": "https://www.moma.org/artists/3228",  
 "\_label": "Joseph Kosuth"  
 }  
 ]  
 }  
}

Parece mais complexo que o exemplo anterior, e de fato é, mas agora temos um arquivo JSON que pode ser lido por humanos e máquinas, e que pode ser utilizado para conectar dados, trazendo muito mais contexto para a informação, em que: *“@context”* é o link para o vocabulário utilizado, no caso Linked Art; “*id*” é o identificador do objeto, o link para o registro no MoMA; “*\_label*” é um rótulo para leitura pelo desenvolvedor; “*type*” é o tipo de entidade; “*identified\_by*” é a propriedade para identificar a obra, recebe um array de objetos que podem ser tanto “*Name*” quanto “*Identifier*”, no caso é um “*Identifier*”com o valor “*One and three chairs*” classificado como “*Title*” e referenciado à definição de “*title*” no vocabulário AAT do Getty. Por fim,“*produced\_by*” é a propriedade que recebe um objeto “*Production*” que tem como propriedade “*carried\_out\_by*” que recebe um array de objetos “*Person*” que tem como propriedade “*id*” o link para o registro do artista no MoMA e “*\_label*” o nome do artista.

O JSON-LD fornece uma maneira de os dados JSON serem interoperáveis na escala da Web (JSON-LD WORKING GROUP, 2020). E, tem como principal intenção “ser uma maneira de usar Linked Data em ambientes de programação baseados na Web, para construir serviços Web interoperáveis e para armazenar Linked Data em mecanismos de armazenamento baseados em JSON” (JSON-LD WORKING GROUP, 2020), ou seja, é o formato ideal para serviços REST.

## JSON-Schema

*JSON-Schema* é um vocabulário que permite a descrição e validação de documentos JSON (DROETTBOOM, 2020). O JSON-Schema é um documento JSON que descreve a estrutura de um JSON, ou seja, é um documento que descreve o que um JSON pode conter, quais são os tipos de dados, quais são os valores permitidos, quais são os valores obrigatórios, entre outras informações. O JSON-Schema também é comumente utilizado para descrever e validar dados de uma API REST.

**Exemplo 3**: Descrição de um objeto com uma propriedade *name* do tipo *string*:

{  
 "type": "object",  
 "properties": {  
 "name": {  
 "type": "string"  
 }  
 }  
}

em que seria possível validar o seguinte JSON:

{  
 "name": "One and three chairs"  
}

Uma das principais características de um documento JSON-Schema é a possibilidade de referenciar outros *schemas*, importar suas definições e reutilizá-las. Essa característica é muito importante para a descrição de dados, pois permite a reutilização de definições de *schemas*, evitando a repetição de código e facilitando a manutenção. A referência é feita através da propriedade “$ref”, que recebe uma URI ou um “*path*” [[2]](#footnote-28) que aponta para o *schema* que será referenciado.

Utilizando o exemplo 2 como base, podemos descrever o JSON utilizando o JSON-Schema e utilizando a propriedade “$ref” da seguinte forma:

**Exemplo 4**: JSON-Schema descrevendo os metadados do Exemplo 2

// object.json  
{  
 "$schema": "http://json-schema.org/schema#",  
 "title": "Human-Made Object Schema",  
 "description": "\_crm:E22\\\_Human-Made\\\_Object\_\nA human-made object, part thereof, or a natural object with value imbued through interaction with human culture.\nSee: [API](https://linked.art/api/1.0/endpoint/object/) | [Model](https://linked.art/model/object/)",  
 "type": "object",  
 "properties": {  
 "produced\_by": {  
 "title": "crm:P108i\_was\_produced\_by",  
 "description": "The `Production` activity which created this object",  
 "$ref": "#/definitions/Production"  
 }  
 },  
 "definitions": {  
 "Production": {  
 "title": "crm:E12\_Production",  
 "description": "The production/creation of the object",  
 "type": "object",  
 "properties": {  
 "carried\_out\_by": {  
 "$ref": "core.json#/definitions/carried\_out\_byProp"  
 }  
 }  
 }  
 },  
 "required": ["@context", "id", "type", "\_label"],  
 "additionalProperties": false  
}  
  
// core.json  
{  
 "$schema": "http://json-schema.org/schema#",  
 "title": "Core Schema Definitions",  
 "description": "Core definitions that are used to compose the Linked Art JSON-LD API",  
 "type": "object",  
 "definitions": {  
 "carried\_out\_byProp": {  
 "title": "crm:P14\_carried\_out\_by",  
 "description": "A reference to a Person or Group which carried out this activity",  
 "type": "array",  
 "items": {  
 "$ref": "#/definitions/PersonRefOrGroupRef"  
 }  
 }  
 }  
}

**Fonte**: Versões resumidas do *object.json* e *core.json* do Linked Art (SANDERSON, 2021)

No exemplo 4 temos a definição de dois arquivos *json*, *object.json* e *core.json*. O arquivo *object.json* descreve o objeto *Human-Made Object Schema* e o arquivo *core.json* descreve as definições que são utilizadas para compor esta e outras classes do sistema. O arquivo *object.json* possui uma propriedade *produced\_by* que é do tipo *Production*, definida no arquivo *object.json* em *definitions*. *produced\_by* é uma propriedade que pode conter um ou mais valores, por isso é do tipo *array* e cada item do *array* é uma referência para um objeto do tipo *PersonRefOrGroupRef*, que é definido no arquivo *core.json* (omitido nesta versão resumida).

## API REST

*API*[[3]](#footnote-30) *REST* é um conjunto de princípios de arquitetura para sistemas distribuídos (FIELDING, 2000). O termo *REST* foi introduzido em 2000 por Roy Fielding em sua tese de doutorado (FIELDING, 2000), e é a abreviação de *Representational State Transfer*, ou seja, transferência de estado representacional. O termo *REST* é utilizado para descrever qualquer interface entre sistemas que utilize o protocolo HTTP para obter dados ou gerar operações sobre esses dados em todos os formatos possíveis, como XML e JSON, por exemplo (FIELDING, 2000). A arquitetura do REST consiste em adicionar camadas de restrições à aplicações:

There are two common perspectives on the process of architectural design, whether it be for buildings or for software. The first is that a designer starts with nothing–a blank slate, whiteboard, or drawing board–and builds-up an architecture from familiar components until it satisfies the needs of the intended system. The second is that a designer starts with the system needs as a whole, without constraints, and then incrementally identifies and applies constraints to elements of the system in order to differentiate the design space and allow the forces that influence system behavior to flow naturally, in harmony with the system. Where the first emphasizes creativity and unbounded vision, the second emphasizes restraint and understanding of the system context. REST has been developed using the latter process. (FIELDING, 2000)

As camadas de restrições são (FIELDING, 2000):

1. **Separação cliente-servidor**: ao separar a lógica dos dados da interface do usuário, melhoramos a portabilidade da interface de usuário em várias plataformas (computador, celular, tablet) e melhoramos a escalabilidade ao simplificar os componentes do servidor;
2. **Stateless**: a comunicação do cliente com o servidor tem que ser sem estado por natureza. Isso significa que cada requisição feita do cliente para o servidor deverá conter toda a informação necessária para a requisição ser compreendida;
3. **Cache**: como a comunicação é sem estado, isso pode acarretar em ineficiência da rede, dessa forma adicionamos o controle do cache para reutilização de informação requisitada anteriormente;
4. **Interface uniforme**: a característica principal que distingue a arquitetura REST de outra arquitetura de rede é a interface uniforme entre componentes. As implementações são dissociadas dos serviços que fornecem, o que incentiva a evolução independente; e
5. **Sistema em camadas**: ao compor a arquitetura em camadas hierárquicas, restringimos os componentes a não irem além de seus escopos.

:::caution TODO

falar sobre api do linked art, principios de design e protocolos

:::

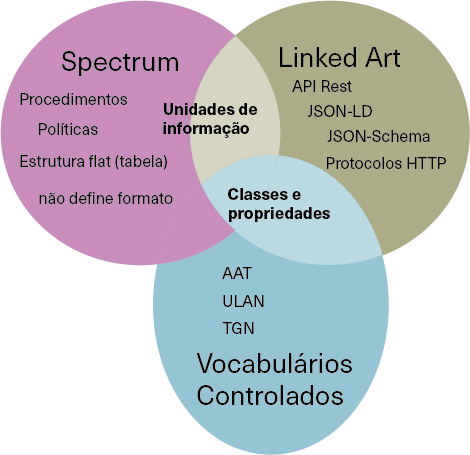
# Interseccoes entre Linked Art e Spectrum

Como vimos nas seções anteriores, o Spectrum define um conjunto de procedimentos e unidades de informação, em que as unidades de informação são bastantes especializadas, como “Object name” ou “Conservation authorization date”, o que por si só já traz contexto. O Spectrum também não determina o formato de registro das informações: se devemos utilizar tabelas em CSV, MYSQL, XML, RDF, ou qualquer outro formato para dados,

Em contrapartida, o Linked Art utiliza o formato JSON, mais especificamente o JSON-LD e define uma estrutura comum que é compartilhada por diversas propriedades e adiciona uma camada semântica com o uso de vocabulários controlados para definição dos conceitos descritos.

Para o mapeamento entre o Spectrum e o Linked Art, selecionamos apenas as unidades de informação do Spectrum, uma vez que os procedimentos são atividades que vão além do registro em si, mas suas informações ainda podem ser representados pelo Linked Art, como veremos mais adiante. Também selecionamos as Classes e propriedades do Linked Art, que necessitam da contextualização de um vocabulário controlado como pode ser visto na Figura 1.

**Figura 1** - Intersecções entre o Spectrum e o Linked Art.



interseccoes

**Fonte:** Elaborado pelos autores.

A imagem acima demonstra a importância do uso de vocabulários controlados para trazer mais contexto para os metadados no Linked Art, como por exemplo a propriedade *identified\_by* do Linked Art é a forma padrão de identificação de qualquer classe definida pelo modelo, como *Object*, *Concept*, *Digital* *Object*, *Event*, entre outras. Ela aceita um array de objetos que podem ser ou *Identifiers* ou *Names* que possuem a propriedade *classified\_as*, um array de objetos *Concept* que representam os conceitos identificados pelo *content* da propriedade:

**Exemplo x**: Identificação de um objeto, demonstra a propriedade *identified\_by* com um array de objetos *Identifier* e *Name* e suas propriedades *classified\_as*.

{  
 "identified\_by": [  
 {  
 "type": "Identifier",  
 "\_label": "Accession Number",  
 "classified\_as": [  
 {  
 "id": "http://vocab.getty.edu/aat/300404621",  
 "\_label": "Accession Number"  
 }  
 ],  
 "content": "M-0821"  
 },  
 {  
 "type": "Name",  
 "\_label": "Title",  
 "classified\_as": [  
 {  
 "id": "http://vocab.getty.edu/aat/300404621",  
 "\_label": "Title"  
 }  
 ],  
 "content": "Rio de Janeiro"  
 }  
 ]  
}

**Fonte**: Elaborado pelos autores. Baseado no Linked Art, descreve a obra Rio de Janeiro de Tarsila do Amaral.

Já o Spectrum define campos especializados para o tipo de identificação que estamos realizando, seja de um objeto (COLLECTIONS TRUST, 2017):

* *Object Name*;
* *Object Number*;
* *Title*;
* *Other Number*.

Ou de uma pessoa (COLLECTIONS TRUST, 2017):

* Person’s reference number;
* Person’s forename;
* Person’s surname;
* Person’s title;
* entre outras.

BEVILACQUA, Gabriel Moore Forell. **SPECTRUM 4.0: O padrão para gestão de coleções de museus do Reino Unido**. São Paulo: Secretaria de Estado de Cultura; Associação de Amigos do Museu do Café; Pinacoteca do Estado de São Paulo, 2014. v. 2 Disponível em: https://spectrum-pt.org/wp-content/uploads/2021/03/Spectrum\_PT\_NET.pdf. Acesso em: 20 maio. 2023.

COLLECTIONS TRUST. **Appendix - information requirements - Spectrum**. 2017a. Disponível em: https://collectionstrust.org.uk/spectrum/information-requirements/. Acesso em: 21 maio. 2023.

COLLECTIONS TRUST. **Object name - Spectrum**. 2017b. Disponível em: https://collectionstrust.org.uk/resource/object-name/. Acesso em: 22 maio. 2023.

COLLECTIONS TRUST. **Object identification information - Spectrum 5.1**. 2017c. Disponível em: https://collectionstrust.org.uk/resource/object-identification-information/. Acesso em: 27 maio. 2023.

COLLECTIONS TRUST. **Person information**. 2017d. Disponível em: https://collectionstrust.org.uk/resource/person-information/. Acesso em: 27 maio. 2023.

COLLECTIONS TRUST. **Introduction do Spectrum**. 2022. Disponível em: https://collectionstrust.org.uk/spectrum/spectrum-5/. Acesso em: 19 maio. 2023.

COLLECTIONS TRUST. **Primary procedures**. 2023. Disponível em: https://collectionstrust.org.uk/spectrum/primary-procedures/. Acesso em: 20 maio. 2023.

COLLECTIONS TRUST. **Spectrum Timeline**. [s.d.]. Disponível em: https://collectionstrust.org.uk/resource/spectrum-timeline/. Acesso em: 20 maio. 2023a.

COLLECTIONS TRUST. **#RethinkingCataloguing**. [s.d.]. Disponível em: https://collectionstrust.org.uk/blog/rethinkingcataloguing/. Acesso em: 20 maio. 2023b.

DROETTBOOM, Michael. **Understanding JSON Schema**. [s.l.] : Space Telescope Science Institute, 2020. Disponível em: https://json-schema.org/understanding-json-schema/. Acesso em: 27 maio. 2023.

FIELDING, Roy T. Representational State Transfer (REST). *Em*: **Architectural Styles and the Design of Network-based Softwares Architectures**. Irvine: Information and Computer Science - University of California, 2000. Disponível em: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\_arch\_style.htm. Acesso em: 3 ago. 2022.

JSON-LD WORKING GROUP. **JSON-LD 1.1 - A JSON-based Serialization for Linked Data**. 2020. Disponível em: https://www.w3.org/TR/json-ld11/. Acesso em: 3 ago. 2022.

LINKED ART. **Linked Art**. 2021a. Disponível em: https://linked.art/. Acesso em: 20 maio. 2023.

LINKED ART. **Community - Linked Art**. 2021b. Disponível em: https://linked.art/community/. Acesso em: 20 maio. 2023.

LINKED ART. **Linked Art Profile of CIDOC-CRM**. 2021c. Disponível em: https://linked.art/model/profile/. Acesso em: 20 maio. 2023.

LINKED ART. **API - Linked Art**. 2021d. Disponível em: https://linked.art/api/1.0/. Acesso em: 21 maio. 2023.

LINKED ART. **Linked Art API Endpoints**. 2021e. Disponível em: https://linked.art/api/1.0/endpoint/. Acesso em: 21 maio. 2023.

MATOS, Alexandre. Nota sobre a tradução do SPECTRUM em Portugal. *Em*: **SPECTRUM 4.0: o padrão para gestão de coleções de museus do Reino Unido**. São Paulo: Secretaria de Estado de Cultura; Associação de Amigos do Museu do Café; Pinacoteca do Estado de São Paulo, 2014. v. 2. Disponível em: https://spectrum-pt.org/wp-content/uploads/2021/03/Spectrum\_PT\_NET.pdf. Acesso em: 19 maio. 2023.

MDN WEB DOCS. **Working with JSON**. 2023. Disponível em: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/JSON. Acesso em: 21 maio. 2023.

SANDERSON, Robert. **Shout It Loud: LOUD - EuropeanaTech 2018 Keynotes**. 2018. Disponível em: https://youtu.be/r4afi8mGVAY. Acesso em: 20 maio. 2023.

SANDERSON, Robert. **object.json**. 2021. Acesso em: 27 maio. 2023.

SPORNY, Manu. **What is JSON-LD?** 2012. Disponível em: https://www.youtube.com/watch?v=vioCbTo3C-4. Acesso em: 22 maio. 2023.

W3C JSON-LD WORKING GROUP. **JSON for Linking Data**. 2014. Disponível em: https://json-ld.org. Acesso em: 3 ago. 2022.

1. (BEVILACQUA, 2014) [↑](#footnote-ref-23)
2. O termo “path” é utilizado para descrever o caminho de um arquivo em um sistema de arquivos. Por exemplo, o caminho do arquivo “index.html” na pasta “public” seria “public/index.html”. [↑](#footnote-ref-28)
3. O termo API ou *Application Programming Interface* é amplamente utilizado para definir interfaces de programação, ou seja, qualquer software com uma função distinta. Existem outras interfaces amplamente utilizadas por desenvolvedores e usuários, como a CLI - *Command Line Interface* e UI - *User Interface*. [↑](#footnote-ref-30)