

**PADRÃO REDFISH® DE GERENCIAMENTO DE SISTEMAS PARA
GERENCIAMENTO DE HARDWARE, SOFTWARE E INTERFACE DE
PROGRAMAÇÃO DE APLICATIVOS (API) NO PADRÃO O-PAS**

***SYSTEM MANAGEMENT STANDARD REDFISH® TO O-PAS STANDARD
MANAGEMENT HARDWARE, SOFTWARE AND APPLICATION PROGRAMING
INTERFACE (API)***

Vitor Bergantini Parede¹

Vanderlei Luiz Daneluz Miranda²

RESUMO

O presente trabalho tem por objetivo descrever a atuação do padrão Redfish, como gerenciador de sistemas, um dos padrões abertos escolhido pela especificação Open Process Automation Standard. O gerenciamento de sistemas é feito por uma API Redfish utilizando comandos padrões CRUD, categorizada como RESTfull, devido a leitura simplificada para humanos, e por ser seguro. O intuito do uso de um padrão aberto de gerenciamento de sistemas aliado a outro padrão aberto de orquestração é entregar para os fabricantes um formato que garante ao usuário final intercambialidade e escalabilidade dos dispositivos (transmissores, DCNs, servidores, PLCs, etc.) e conseqüentemente, aumento da vida útil de plantas no âmbito da automação industrial. Por isso, o Padrão O-PAS procura utilizar normas e padrões já existentes no mercado, sendo conhecido como um padrão de padrões. A modalidade desta pesquisa é bibliográfica e será apoiada no material disponibilizado pelo Open Group, consórcio global de soluções em padrões de TI, em seu website, palestras, revistas de inovação tecnológica e no próprio manual do padrão Redfish disponível no website da DMTF.

Palavras-chave: Redfish. Automação Industrial. Gerenciamento de sistemas. O-PAS.

¹ Graduação em Engenharia Elétrica no Centro Universitário UNIFAFIBE de Bebedouro, SP. E-mail: vparede@hotmail.com

² Professor Mestre, no Centro Universitário UNIFAFIBE de Bebedouro, SP. E-mail: vandmir@gmail.com

ABSTRACT

This paper goals describe how Redfish standars act, has a system manager, one of several open standards choseds by the Open Process Automation Standars. The system management are made by and Redfish API using CRUD commands, is RESTfull, because human readable easeness and safety. The purpose to use a system management open standard allied to an orchestration open standard is show a standard to suppliers ensure devices interchangeability and scalability (transmitter, DCNs, servers, PLCs, etc.) and grow plant lifecicle on industrial automation. O-PAS standard work with norms and standards already existing on the market, enabling greater integration between suppliers and users, making hardware and software replacement faster and simple. This bibliography research will be supported by Open Group papers, a global consortium of TI solutions standards, on its website, lectures, technological innovation magazines and in the Redfish manual, available on DMTF website.

Keywords: Redfish. Industrial Automation. System Management. O-PAS.

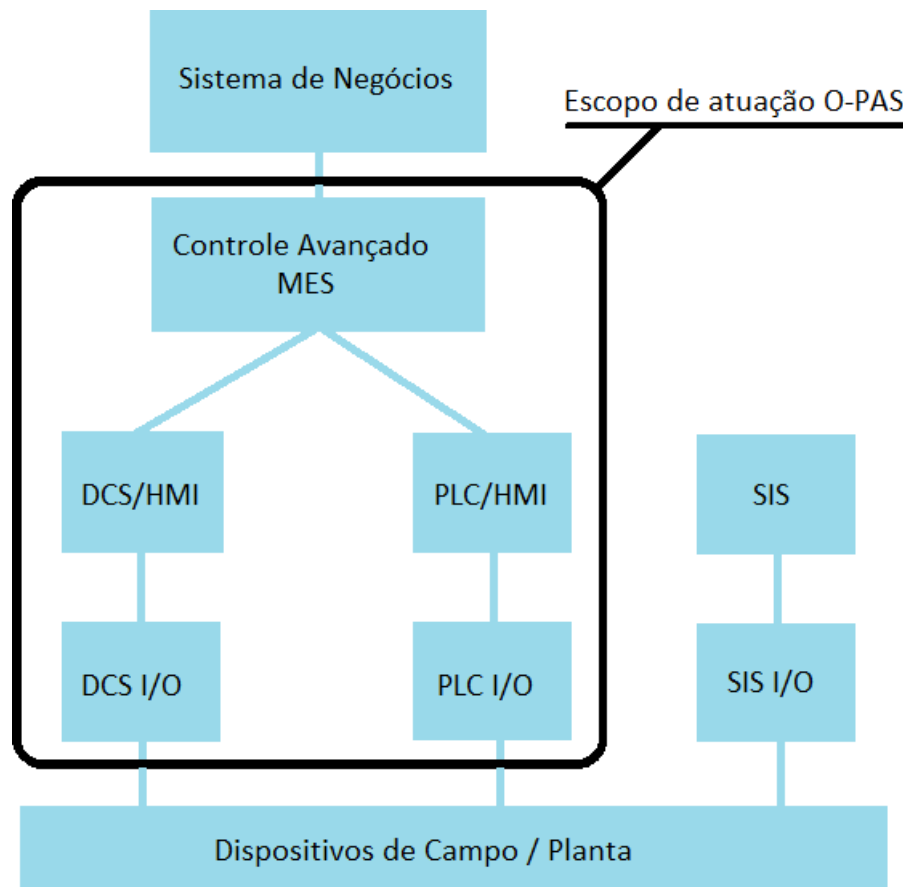
1 INTRODUÇÃO

Diversos segmentos da indústria, como, Gás e Petróleo, Fármacos, Papel e Celulose, Químicos e Mineração, buscam diariamente solucionar problemas e otimizar seus processos. Com o término da vida útil de equipamentos na indústria é necessário fazer a sua substituição, um processo complexo, demorado e custoso. Há uma preocupação constante das empresas que se beneficiam da automação industrial de reduzir o custo envolvido, sendo, entretanto, algo de difícil solução. Dessa forma, um grupo de usuários do setor industrial criou o *Open Process Automation Forum* (OPAF), iniciando uma solução para esse problema para os próximos anos. O resultado desse esforço conjunto deu origem a uma especificação: o *Open Process Automation Standard* (O-PAS). O Padrão O-PAS procura utilizar normas e padrões de fato existentes e consolidados no mercado, possibilitando uma maior integração entre fornecedores e usuários e, ao mesmo tempo, tornando mais simples e ágil a substituição de *hardware* e *softwares* em plantas industriais (THE OPEN GROUP, 2018b).

A Figura 1 mostra o escopo do padrão O-PAS. O padrão irá englobar funções

de controles híbridos e contínuos, a visualização e operação de sistemas de controle, estratégias avançadas de controle e gerenciamento de operações de manufatura e atividades que necessitem operar em complexos de produção em tempo real (THE OPEN GROUP, 2018b). O Fórum utiliza o O-PAS para atuar dentro do Sistema de Execução de Manufatura (*Manufacturing Execution System – MES*), dos Nós de Controle de Sistema (*Distributed Control System – DCS*), dos Controladores Lógicos Programáveis (*Programmable Logic Controller - CLP*), nas Interfaces Homem-Máquina (*Interface Human-Machine – IHM*) e das Entradas e Saídas (*Input and Output - I/O*) de PLC e DCS.

Figura 1 - Escopo de atuação O-PAS



Fonte: Open Group (2018)

O foco do Fórum recai sobre padrões e práticas de negócio necessárias para conseguir interoperabilidade, modularidade e portabilidade de componentes de *software* e *hardware*. Esses atributos trazem vantagem ao negócio, oferecem reuso de componentes, maior facilidade de implementação de novas tecnologias e uma

melhoria no gerenciamento da obsolescência (THE OPEN GROUP, 2018b).

A intenção do Fórum é disponibilizar um verdadeiro desenvolvimento de execução de controle distribuído para operações de manufatura e soluções de controle, com suporte para atualizações *zero downtime* (tempo de parada zero), melhoria na performance contínua e integração de soluções para múltiplos fornecedores. O OPAF avalia e incorpora padrões abertos, reunindo em um conjunto de documentos que orienta fornecedores e consumidores finais sobre como deverão ser compostos e como atuarão *hardware* e *software*. São 7 partes, cada uma se refere a um conjunto de padrões e regras para garantir interoperabilidade, modularidade e portabilidade à planta (THE OPEN GROUP, 2020a).

O presente trabalho tem como foco o estudo do padrão aberto *Redfish*, definido pela organização não governamental *Distributed Management Task Force* (DMTF), adotado como solução aberta para as funções de gerenciamento de sistema em plantas de automação pelo padrão OPA-S, o Redfish define uma *Application Programming Interfaces* (API) usando códigos (comandos e status) HTTPS transferidos via *Representational State Transfer* (Restfull) (THE OPEN GROUP, 2020a). Segundo o OPA-S, os recursos a serem gerenciados da planta são mapeados por meio de um modelo de dados comum, definido por *schemas* que representem de forma textual as propriedades que definem tais recursos como por exemplo, memória disponível, processador, versão de *firmware*, quantidade de entradas e saídas (I/O) disponíveis e ocupadas, entre outros dados que sejam específicos dos dispositivos usados na planta (THE OPEN GROUP, 2020b).

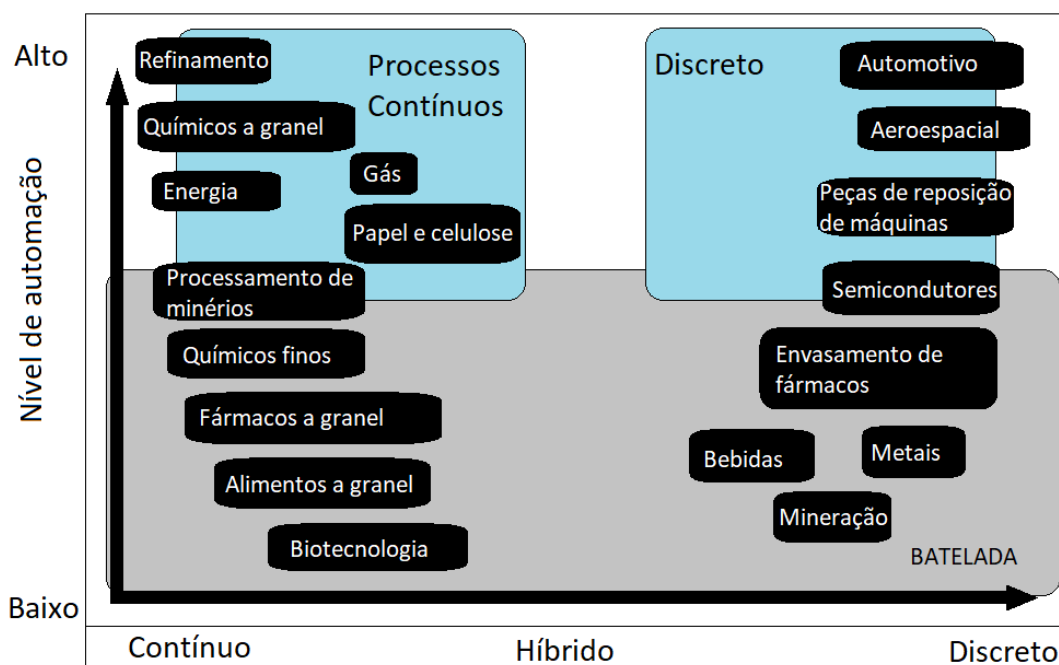
2 REQUISITOS PARA GERENCIAMENTO DO SISTEMA

Visto que diferentes segmentos possuem diferentes níveis de controle, o Fórum divide as diferentes áreas em uma escala de processos de contínuos a discretos, conforme Figura 2. O padrão responderá, inicialmente, para os processos que estejam na faixa do mais contínuo à processos híbridos (THE OPEN GROUP, 2018b).

O padrão mantém o objetivo inicial de interoperabilidade, modularidade e portabilidade de software e hardware, independente do fornecedor e tipo de controle. Para isso é necessário a padronização das ferramentas de gerenciamento para que seja possível gerenciar com facilidade e clareza dentro do processo de controle. Para

software será adotado o uso de APIs e modelos de dados, e para hardware será adotada uma padronização de funcionalidade e forma (THE OPEN GROUP, 2018b).

Figura 2 - Análise de nível de automação por segmento



Fonte: Open Group (2018)

O *revamp* de um sistema de controle dentro de uma planta de processo contínuo, é um grande desafio, devido ao tipo de processo que não permite paradas abruptas ou mesmo paradas programadas que sejam longas. Plantas de processo contínuo requerem curtos prazos de manutenção e atualização, processos como controle e monitoramento de temperatura por exemplo, ao considerar uma troca offline. Uma substituição online por etapas está ligada ao seguinte processo, iniciando a partir da instalação da nova Interface Homem Máquina (IHM) que atuará paralela a atual (THE OPEN GROUP, 2018b).

Após treinamento total dos operadores é necessário descomissionar a antiga camada IHM e iniciar a substituição do antigo *Distributed Control System* (DCS); finalizando o desenvolvimento, as aplicações lógicas e os novos gráficos do IHM dentro do DCS, é possível iniciar a fase de implementar os novos controladores (será necessário gerenciar bem o espaço devido a coexistência dos antigos e novos controladores simultaneamente). Ao final, serão descomissionados os antigos controladores (THE OPEN GROUP, 2018b).

Adotando o padrão OPAS, este processo terá maior facilidade devido a pontos como, padronização das IHM (permitindo melhorias independentes dos seus controladores), eliminando a necessidade de duas IHMs simultâneas, padronização das APIs, não haverá necessidade de se manter antigos e novos controladores simultâneos no novo DCS, padronizar entradas e saídas (I/O) e permitir atualização dos controladores independentemente dos I/Os, reduz os riscos que envolvem uma transição e atualização (*cutover*) da planta (THE OPEN GROUP, 2018b).

Para que um sistema seja capaz de realizar todo o gerenciamento do processo, realizar um *cutover* e iniciar a planta em um curto prazo (*offline* ou *online*) o OPAF institui uma série de requisitos de sistema que especificam como é necessária sua construção, esses requisitos não especificam como qualquer sistema deverá ser capaz de atuar, mas como um sistema deverá atuar para os componentes do padrão O-PAS (THE OPEN GROUP, 2018a).

Os requisitos podem ser identificados no documento *Requirements for an Open Process Automation™ Standard* separados por ID. Os IDs para requisitos de sistemas são: REQ-011, REQ-012, REQ-026, REQ-054, REQ-091, REQ-148 e REQ-215. De forma geral, os requisitos de sistemas são construídos para que os componentes que sejam construídos dentro da especificação OPAS, permitam:

- a) o sistema reduzir os *loops* de leitura (tornando cada *loop* de controle mais rápido);
- b) sistemas O-PAS serão capazes de acrescentar novas I/Os sem que seja necessário um novo sistema (permitindo escalabilidade da planta);
- c) o sistema terá suporte ao padrão IEC 61131-3, a qual define metodologias de construção de lógica estruturada e modular e as linguagens que serão usadas, dentre *Structures Text*, *Instruction list*, *Ladder*, *Function Block Diagram* e *Sequential Flow Chart*;
- d) os componentes deverão suportar o *Service-Oriented Architecture* (Serviço Orientado a Arquitetura - SOA);
- e) os componentes deverão ser intercambiáveis, não deverão ser fixados pela sua arquitetura, deverão permitir flexibilidade;
- f) a programação de controle do engenheiro deverá ser desenvolvida em um padrão que tenha um método consistente para acesso as I/Os, histórico e outros dados;

- g) o sistema deverá ter suporte a estabilização de padronização do controle lógico entre diferentes controladores de diferentes fornecedores para ele ou sistemas diferentes (THE OPEN GROUP, 2018a).

3 GERENCIAMENTO DE SISTEMA NO ESCOPO DA AUTOMAÇÃO

No escopo da automação de controle de processos, qualquer que seja seu controle, é possível a seguinte divisão:

- a) *system management* (gerenciamento de sistema);
- b) *network management* (gerenciamento de rede);
- c) *application management* (gerenciamento de aplicações).

Como define Barreto Filho (2016), gerenciamento de redes se define pelo controle de qualquer ação e pelo monitoramento dos recursos da rede, por meio de informações e diagnósticos.

Já o gerenciamento de aplicações, pode ser definida como gerenciamento da vida útil e da operação de aplicações, sua manutenção e atualização, controle da versão de operação (VMware, c2021).

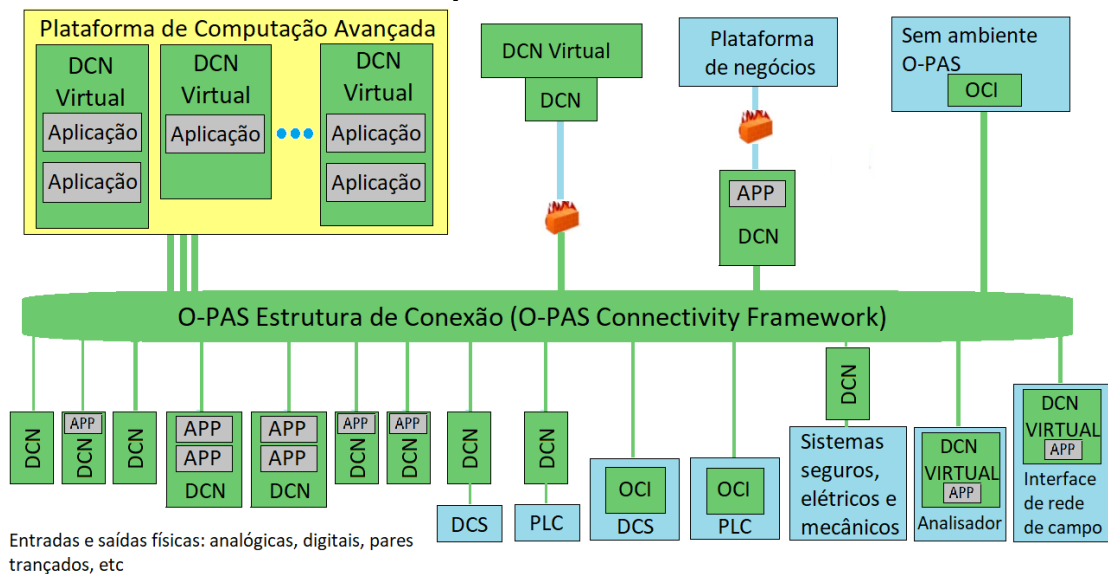
O gerenciamento de sistemas, que é o objetivo deste trabalho, foi inicialmente pensado e desenvolvido para soluções de Tecnologia da Informação (TI), buscando facilitar o acesso a informações e gerenciar aplicações, ativos de rede, administrar a rede, tratar da segurança da rede, automatizar *backups*, atualizar antivírus e tudo relacionado ao bem-estar da rede (PC&Network Download, 2021).

A automação aproveita e modifica o gerenciamento de sistemas, trazendo para a Tecnologia Operacional (TO) como uma nova forma de gestão e organização. No âmbito da integração TI & TO, o gerenciamento de sistemas é definido como como um sistema capaz de administrar e gerenciar a vida útil dos componentes de *hardware* e *software* na planta, desde a implementação até seu descomissionamento, sendo capaz de identificar a versão de firmware, quantidades de I/O disponíveis, memória, dispositivos integrados (acionadores, válvulas, transmissores etc.), status de funcionamento e outros (THE OPEN GROUP, 2020b).

Desta forma, a automação de processos envolve uma variedade de computadores e nós de rede permitindo que aplicações possam controlar os processos.

Conforme mostrado na Figura 3, o gerenciamento de sistemas atua nos nós de dispositivos em *Distributed Control Nodes* (DNC) e *Programmable Logic Controller* (PLC), em plataformas de computação avançadas e, apesar de não ser demonstrado na imagem, nos nós de redes (THE OPEN GROUP, 2020b). Na Figura 3, todos os campos de cor verde referem-se a componentes atuando conforme o padrão O-PAS, enquanto os campos em cores azuis referem-se aos dispositivos que não possuem padronização.

Figura 3 – Exemplo de configuração de componentes atuando com padrão O-PAS



Fonte: Open Group, 2018

3.1 *Suite* de Aplicações de Gerenciamento de Sistema

Dentro da automação de processos, o conceito de *suite* de aplicativos é entendido como um conjunto de aplicações/serviços que auxiliam o gerenciamento (sistema, rede e aplicativos de blocos de função).

O Fórum determina um *suite* para o padrão, relacionando funções e aplicações que utilizam o *Redfish* para realizar seu gerenciamento em termos físicos e virtuais, conforme descrito abaixo:

- a) Identificação e descobrimento de nós: O gerenciamento de sistema deve suportar o sistema de *plug-and-play*, buscando automaticamente nós virtuais e físicos, garantindo suporte às tarefas do sistema (Open Group, 2018b).
- b) Inventário de nós e componentes: Realizar o inventário de *hardware* físico e virtual é vital para um bom funcionamento em um sistema de automação de processos, deve contemplar o gerenciamento da segurança, manutenção, capacidade e vida útil. O gerenciamento é capaz de identificar *firmware/hardware* (versão, fabricante, atualização, memória, entradas e saídas) e listagem dos subcomponentes de *hardware* (Open Group, 2018b).
- c) Configuração de *hardware*: Gerenciar mudanças de *hardware* ocasionadas pela atualização de *firmware*, verificando mudanças relacionadas a controles, estado, data e hora, telemetria e as políticas de segurança e privacidade do fabricante (Open Group, 2018b).
- d) Tarefas de *backup* e configuração dos nós de gerenciamento de sistema: Garantir a capacidade de gerenciar e registrar a vida útil durante mudanças de *hardware*, garantindo a capacidade de auditar, rastrear e repor. A interface do gerenciamento de sistemas deve ser capaz de registrar e armazenar essas alterações e tarefas, podendo carregá-las ao nó de *hardware* (Open Group, 2018b).
- e) Monitoramento de sistema e captura de eventos: Todo e qualquer evento relacionado ao nó virtual ou ao *hardware*, é registrado e enviado ao setor de OT para registro e verificação de alarmes, estado das aplicações, telemetrias e outros (Open Group, 2018b).
- f) Aplicações de análise: No escopo de sistemas de automação de processo, a análise de tendências de eventos em *hardware* é feita por meio de aplicações

em situações como alertas de manutenção preventiva, capacidade de gerenciamento, carga e descarga, e outros (Open Group, 2018b).

- g) Operações: A aplicação de operação deve atuar enviando requerimentos e comandos para um servidor, ou até mesmo automatizando uma tarefa para o topo da sequência de tarefas e requisições, como em um *patch* de *firmware* (Open Group, 2018b).
- h) *Patches* de *firmware*: Atualizações de comandos, informações de estado, tarefas e outras ações são realizadas como em atualizações de um *firmware*, de forma que o servidor do gerenciamento de sistemas envie tarefas e informações como *patches* de atualização, automatizando atualizações como uma aplicação de orquestramento (Open Group, 2018b).
- i) Análise e segurança auditável: O padrão deve utilizar as normas e controles de segurança e controle auditável conforme a *Industrial Automation and Control Systems* (IACS), para rastreamento, armazenamento de dados, mantendo o histórico em um *data lake* (Open Group, 2018b).

O Quadro 1 traz um resumo deste suite de aplicação descrito para o gerenciamento de sistemas dentro da especificação O-PAS, mostrando os principais serviços e os “alvos” que devem ser atacados.

Quadro 1 - Funções executivas pelo “Suite” da Aplicação de Gerenciamento de Sistemas

| SERVIÇOS | ALVO |
|----------|------|
|----------|------|

| | |
|--|---|
| <ul style="list-style-type: none"> ● Instalar ● Configurar ● Detectar ● Troubleshooting ● Reparar ● Manter | <ul style="list-style-type: none"> ● Hardware O-PAS ● Sistemas Operacionais ● Virtualização ● Containers ● Mecanismos de execução ● Redes ● Outros softwares |
|--|---|

Fonte: Autoria Própria

3.2 APIs do Gerenciamento de Sistema

As APIs de gerenciamento de sistema, são capazes de acessar funções do gerenciamento de sistemas, dados do gerenciamento de software e hardware (Open Group, 2018b).

Há dois mecanismos de comunicação previstos para atuação das APIs:

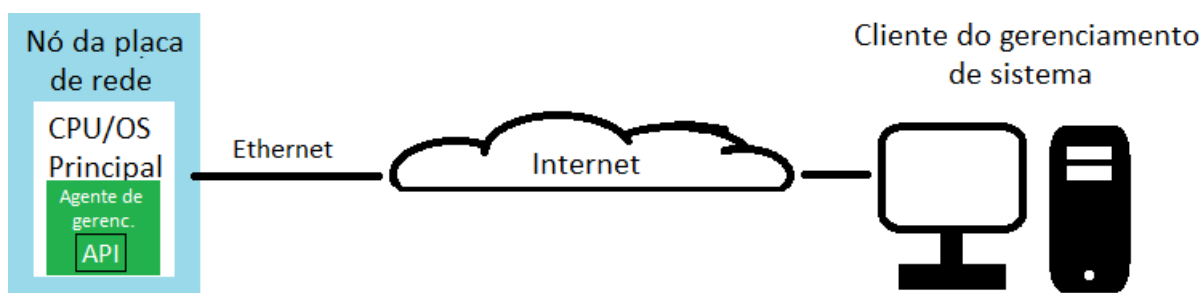
- a) Sessão Cliente-Servidor, onde o *hardware* é o servidor e a aplicação de gerenciamento de sistema é o cliente;
- b) Mecanismo *Pub/Sub* onde a aplicação do gerenciamento de sistemas se registra para eventos específicos no *hardware*.

No que se refere ao gerenciamento de nós de *hardware*, há duas classes previstas:

- a) nó sem *Baseboard Management Controller* (BMC), onde a substituição parcial dos componentes de *hardware* se torna impossível, necessitando a troca do nó inteiro (neste caso ainda haverá a mesma requisição de dados para a API de gerenciamento de sistemas) (Open Group, 2018b);
- b) *hardware* de *desktop* e plataforma de servidor, conhecidos como dispositivos de prateleira, ou, *Commercial Off-The-Shelf* (COTS), permitindo uma facilidade na troca, substituição, personalização e escalabilidade do *hardware* com BMC. Isso causará um problema de padronização, exigindo da API a adaptação necessária para informar dados sensíveis como memória, entrada e saída, *firmware* e outros (Open Group, 2018b).

A Figura 4 representa a comunicação, via Ethernet, de um servidor que esteja com uma API *Redfish* responsável pela busca de dados solicitados por um cliente via Internet, a comunicação cliente-servidor ocorre em um nó sem BMC.

Figura 4 – Comunicação cliente-servidor em nó sem BMC



Fonte: Open Group, 2020

4 PADRÃO REDFISH DMTF

O padrão aberto utilizado para definir a API de gerenciamento de sistema é definido, desenvolvido e distribuído pela organização *Distributed Management Task Force* (DMTF). O *Redfish* define um protocolo, um modelo de dados comum (*schema*) com API de ações, como modelo de dados:

- usa o protocolo HTTPS no formato *JavaScript Object Notation* (JSON) ou YAML;
- apoia-se em *schemas* no formato de texto legível por humano;
- pode ser usado igualmente por aplicações, Interfaces gráficas e *scripts*;
- extensível, seguro e interoperável.

No *Redfish*, cada URL representa um recurso, um serviço ou uma coleção de recursos, uma entidade ou algum outro constructo. Tais recursos são expostos por meio de *schemas* que os clientes podem usar para determinar a semântica de determinados recursos. Assim, todos os recursos estão vinculados através de um serviço (representado por um ponto de entrada no *schema*, o *root*, definido pela URL: `/redfish/v1`). Por exemplo, um aplicativo cliente que deseje pesquisar o número de série de um sistema em Python poderia usar a seguinte seção de código:

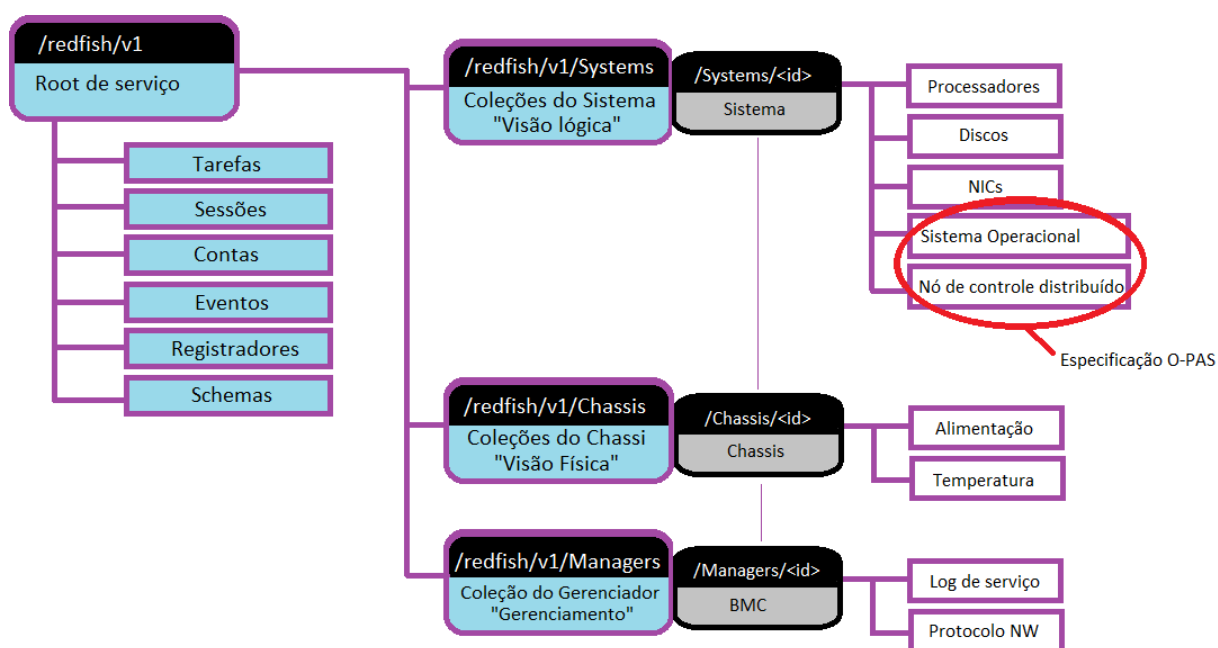
```

rawData = urllib.urlopen('http://192.168.1.135 /redfish/v1/Systems/1')
jsonData = json.loads(rawData)
print( jsonData['Número de Série'] )

```

A figura 5 mostra a hierarquia usada no *schema* a partir do *root* de serviço, demonstrando o formato que a API atua na busca de recursos e onde atua a especificação O-PAS.

Figura 5 – Redfish Schema



Alguns pontos fortes que justificam o uso do padrão *Redfish* na API de gerenciamento de sistema, são:

- Comunicação cliente-servidor através de http *RESTful*;
- Troca de dados e comandos através do formato *JavaScript Object Notation* (JSON);
- Interoperabilidade através do uso de APIs em *Ain't Markup Language* (YAML) e *Common Schema Definition Language* Odata (CSDL Odata).

4.1 *RESTful*

Ser *RESTful* significa que o sistema, ou a API, será capaz de aplicar as definições de *Representational State Transfer (REST)*, tornando-o um sistema claro e padronizado.

O padrão *Redfish* além de ser *RESTful*, atua utilizando os ideais do *CRUD*, *Create – Read – Update – Delete* o que torna a linguagem mais humana e de fácil compreensão. Os comandos *CRUD* mais utilizados no padrão *Redfish* são: *POST*, *GET*, *PUT & PATCH*, *DELETE* e *HEAD*.

A Figura 6 exemplifica a comunicação da API com o cliente utilizando os comandos *CRUD* padrão em comunicação via protocolo HTTP, ambos atuando com o padrão *Redfish*.



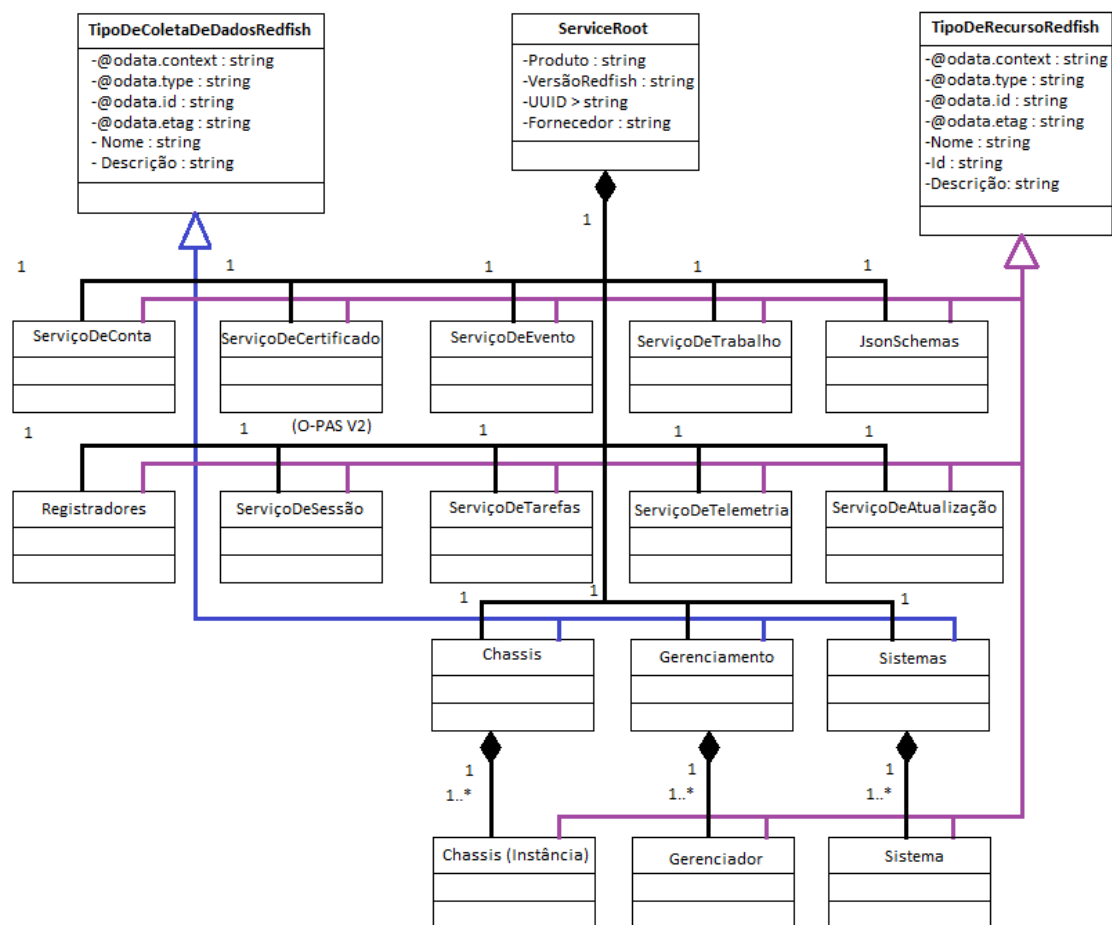
Fonte: Open Group, 2020

4.2 JSON E REDFISH SCHEMAS

JavaScript Object Notation (JSON) é um padrão aberto utilizado para troca de dados entre sistemas em formato simples e rápido de leitura e escrita por humanos, com base em linguagens derivadas da família C, como C, C++, C#, Java, JavaScript, Perl, Python e outras (JSON ORG).

O O-PAS utiliza os *schemas* definidos pelo padrão JSON dentro do padrão *Redfish* para buscar dados em “coleções” identificando diversos dados como temperatura, memória disponível, entradas e saídas disponíveis, versão do *firmware*, *chassis* entre outras informações que sistema precisa para gerenciar e entregar ao usuário, a Figura 7 demonstra o *schema* do *service root* para a especificação O-PAS.

Figura 7 – Service Root JSON schema



Fonte: O-PAS, 2020

Cada resposta da API *Redfish* consiste em dados contendo as propriedades de um recurso definida pelo *schema* do recurso. A descrição completa dos recursos previstos para o Gerenciamento de Sistema na automação pode ser encontrada no capítulo 5.4.2 *Redfish Resources* da Parte 5 do padrão OPA-S. O quadro 2 mostra o conteúdo de uma propriedade relacionada ao recurso.

A especificação O-PAS define 3 perfis para as APIs *Redfish*:

- OSM-001 – Descreve a funcionalidade da base do gerenciamento de sistemas para um nó que tenha um agente *Redfish* atuando no processador principal;
- OSM-002 – Descreve a funcionalidade da base do gerenciamento de sistemas no nó de *hardware* que esteja equipado com um BMC;
- OSM-003 – Atua como o OSM-001, com o acréscimo de função de fornecer dados do sistema operacional (SO), incluindo métricas e o DCN.

Quadro 2 – Propriedade x Recurso

| Coluna | Propósito |
|----------------------|--|
| <i>Property Name</i> | Nome da propriedade (formato JSON) |
| <i>Type</i> | Booleano, número, <i>string</i> ou objeto |
| <i>Attributes</i> | Define se a propriedade é <i>read-only</i> ou <i>read-write</i> |
| <i>Descriptions</i> | Descrição da propriedade (copiada diretamente do campo “ <i>Description</i> ” do <i>schema</i>) |

Fonte: Autoria Própria

Os recursos não se limitam aos demonstrados no Quadro 2, mas também incluem outros definidos pelos perfis OSM-001, OSM-002 e OSM003. O foco de desenvolvimento está no perfil OSM-003 e nos recursos para os DCNs.

4.3 SEGURANÇA

Para garantir a segurança, o padrão *Redfish* trabalha com encriptação de dados e autenticação. A proteção dos dados transmitidos é definida pelo protocolo *Transport Layer Security* (TLS), na atual versão do padrão *Redfish* é suportada a versão v2.0 do protocolo TLS, com as recomendações presentes no documento RFC7525 (DMTF, 2015).

A segurança do *Redfish* dentro da especificação O-PAS está presente no *hardware*, no nó do servidor *web* e na própria API na comunicação entre cliente e servidor (O-PAS, 2010b).

Alguns dos serviços que a API *Redfish* deve ser capaz de oferecer e garantir, de acordo com o padrão O-PAS, são:

- a) Gerenciamento de identidade, gerenciar os acessos a servidores e recursos;
- b) Privilégios e regras, gerenciar os acessos mapeando identidade (autorizações);
- c) Contas, gerenciar contas capazes de acessar o servidor;
- d) Credenciais, gerenciar o nível de autorização, interações de acordo com a conta / identificação;
- e) Certificados, registro de certificado X.509 por sessões TLS;
- f) Acessos, gerenciamento baseado em privilégios para contas servidores.

5 REDFISH E ORQUESTRAÇÃO

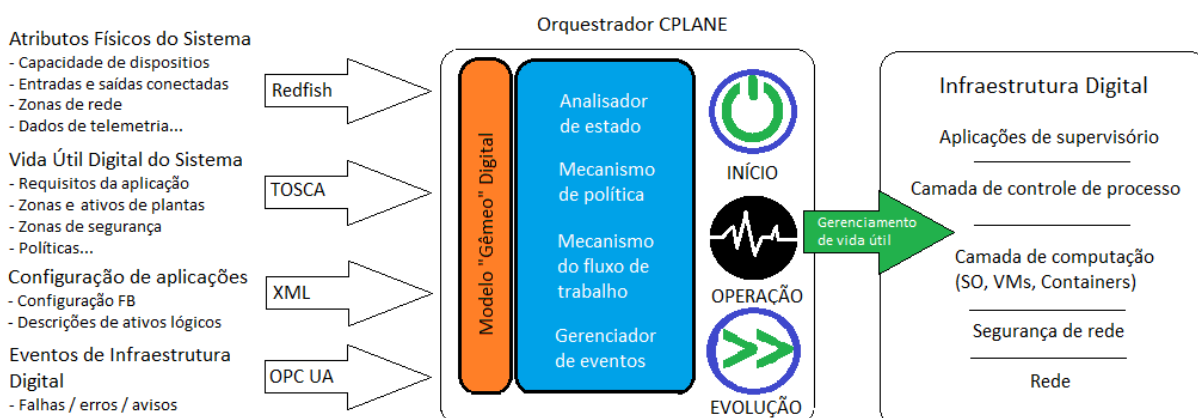
Atualmente a especificação O-PAS se encontra na versão 2.1 que foi anunciada durante o evento digital *Open Digital Standards – Virtual Event* em 21 de Julho de 2021. No primeiro dia do evento o CEO do *Open Group*, Steve Nunn, na companhia de representantes de empresas como *CPLANEai*, *Schneider Eletrics* e *ExxonMobill* debateram e demonstraram a versão 2.1 da especificação, chamando-a de “o padrão de padrões” e acreditam que a especificação fará com que, no futuro, IT e OT estejam no mesmo patamar de desenvolvimento (OPEN GROUP, 2021).

Para que isso ocorra, o padrão O-PAS conta com diversos padrões, sendo o *Redfish* um dos pontos chaves quando aliado a uma orquestração.

A *CPLANEai* realizou um *proof-of concept* (equivalente a prova de conceito), aplicando partes da especificação O-PAS em uma planta desde seu comissionamento.

A demonstração uniu cerca de 14 máquinas distribuídas em 2 *datacenters*, sendo um em Nova Iorque e um na Califórnia (demonstrando uma topologia distribuída, como deve ser). Das 14 máquinas, 13 atuaram como DCNs e 1 como uma plataforma computação avançada, todas possuem componentes internos de diversos fabricantes, algumas com processadores híbridos (parte intel X86 e parte ARM), demonstrando a capacidade de tornar os componentes intercambiáveis. Abaixo a Figura 8 resume a relação entre a orquestração realizada e o papel do *Redfish*, chaves essenciais para o sucesso deste *proof-of concept* (CPLANE.AI AND EXXONMOBILL, 2020).

Figura 8 – Prova de conceito com orquestrador CPLANE



Fonte: Cplane.ai and Exxonmobill, 2020.

6 CONSIDERAÇÕES FINAIS

O padrão O-PAS traz uma solução para a automação industrial sem comparativos, entregando uma facilidade comum para o meio caseiro (o conhecido *plug-and-play* comum em produtos *Microsoft*, por exemplo). Uma das chaves que tornam possível a integração entre fabricantes, entregando ao usuário final produtos intercambiáveis e escaláveis, é a interação entre o padrão *Redfish* no gerenciamento de sistemas e padrões de orquestramento (como o padrão TOSCA).

Enquanto o padrão *Redfish* é responsável por toda a verificação e segurança das informações da planta, realizando a virtualização da planta física, entregando ao usuário dados diversos, alarmes, levando ordens e tarefas, o orquestrador é responsável por casar o trabalho das APIs de gerenciamento com a comunicação sem fio, ou mesmo cabeada, entre usuários (operadores, gestores, responsáveis pela manutenção), dispositivos de diferentes níveis (desde o chão de fábrica até armazenamento de dados em nuvem) e o que mais estiver se comunicando com a planta. Também dentro das responsabilidades do casamento entre gerenciamento de sistemas e orquestramento, está presente o comissionamento, este que se mostra muito eficiente devido o orquestrador, capaz de comissionar toda uma planta em poucas horas, até mesmo minutos (5 minutos no caso do *proof-of concept* da CPLANE.AI).

O uso do *Redfish* entrega benefícios como economia de tempo, controle de ativos e recursos, intercambialidade de dispositivos, escalabilidade da planta, confiabilidade de dados, virtualização completa da planta (permitindo o melhor gerenciamento dos ativos e recursos), gerenciamento de acessos, controle sobre o acesso e compartilhamento de dados e segurança.

REFERÊNCIAS

THE OPEN GROUP. **O-PAS Standard, Version 2.0:** Part 1 – Technical architecture overview (informative). Berkshire UK, 2020a. 49 p.

THE OPEN GROUP. **O-PAS Standard, Version 2.0:** Part 5 – System Management. Berkshire UK, 2020b. 177 p.

THE OPEN GROUP. **Requirements for an Open Process Automation standard.** Berkshire UK, 2018a. 27 p.

THE OPEN GROUP. **The Open Process Automation business guide:** Value proposition and business case for the Open Process Automation standard. Berkshire UK, 2018b. 50 p.

FILHO, Barreto. **Tudo sobre gerenciamento de redes.** Citta telecom, 2016. Disponível em: < <https://www.cittatelecom.com.br/2016/06/gerenciamento-de-redes/> >. Acesso em: 07 de nov. de 2021.

APPLICATION MANAGEMENT. VMware, c2021. Disponível em: < <https://www.vmware.com/topics/glossary/content/application-management> >. Acesso em: 07 de nov. de 2021.

SYSTEMS MANAGEMENT – WHAT IS IT AND A FULL OVERVIEW. PC & Network Downloads, 2021. Disponível em: < <https://www.pcwldd.com/systems-management#wbounce-modal> >. Acesso em: 16 de nov. de 2021.

AHAMED, Rafiq et al. **Redfish Specification.** Versão 1.12.0. Portland: DMTF, 2015, 186 p.

INTODUCING JSON. JSON ORG, [s.d.]. Disponível em: < <https://www.json.org/json-en.html> >. Acesso em: 21 de nov. de 2021.

THE OPEN GROUP OPEN DIGITAL STANDARDS VIRTUAL EVENT. OPEN GROUP, 2021. Disponível em: < <https://blog.opengroup.org/2021/07/27/the-open-group-open-digital-standards-virtual-event-july-19-21-2021-highlights-blog> >. Acesso em: 10 de nov. de 2021.

CPLANE.AI AND EXXONMOBIL. **Orchestration of an Open Industrial Control System.** Silicon Valley, 2020. 11 p.