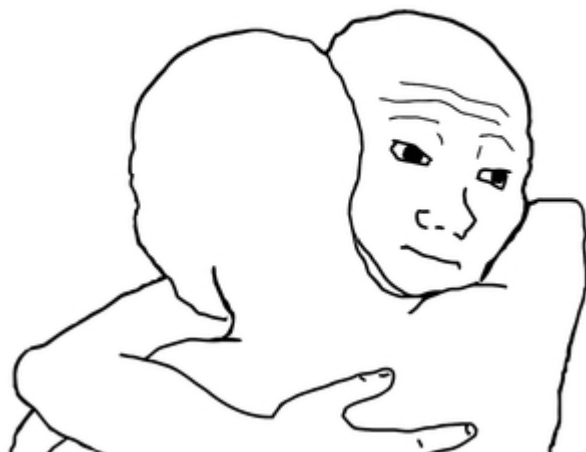


## SAP Netweaver Gateway – Do que se trata essa parada?

19/01/2015 09:00

HANA, SAPUI5, Gateway, Lumira, HANA, River, Fiori, Cloud, HANA, HANA, HANA, etc, etc, etc... Tanta coisa nova, não é mesmo? Quem trabalha com SAP já deve estar careca de saber que de tempos em tempos aparece uma nova avalanche de nomes e siglas e conceitos e slides e guias e livros e... um bando de desenvolvedores perdidos sentindo-se ultrapassados.

I KNOW THAT FEEL BRO



Recentemente eu tive um pouco de contato com uma das ferramentas dessa nova onda de produtos. Fucei um pouco no SAP Netweaver Gateway e vim contar um pouco pra vocês do que se trata essa parada, para que você não fique cambaleando perdido no escritório.

Apesar do Gateway parecer uma coisa extremamente recente, ele já deu as caras no mundo SAP há pelo menos uns 2/3 anos. Sabe como é, leva tempo para as coisas ficarem um pouco mais robustas e terem uma utilização maior em clientes. Particularmente enxergo uma grande utilização do Gateway no futuro, já que o negócio é beeeem firmeza.

Mas antes eu preciso preparar você, caro leitor, para entender os motivos que levaram a SAP a criar essa ferramenta. Por conta disso, antes de explicar a ferramenta, vamos falar um pouco sobre interfaces com usuário.

---

### User Interfaces

Todo mundo aqui já olhou para alguma tela do SAP e pensou “**nossa, que lixo**”. O maior mérito das telas da SAP Gui atuais é a velocidade, tanto de desenvolvimento quanto de uso. Mas ela são feias de dar dor, parecem extremamente ultrapassadas. O ABAP puro dispõe do Webdynpro e BSPs para tentar gerar alguma coisa menos bizarra, mas ambas as tecnologias não são muito populares. O que mais se cria é aquela boa e velha tela 9000 e com aquele ALVzão gigantesco cheio de botões e ícones que não explicam nada da ação daqueles botões (eu já vi um Refresh com o ícone de um cara empurrando caixas).

A questão é que a experiência do usuário com outras tecnologias tem mudado bastante. Não é mais uma questão de comodidade acessar um sistema empresarial por um celular ou tablet, é uma **necessidade**. O design de aplicativos em geral tende a interfaces extremamente simples, mas com uma inteligência robusta e complexa por trás.

Pense em como você fazia antigamente para comprar um ingresso de cinema: você precisava arranjar um jornal com os horários e dias, ir até o cinema, enfrentar uma fila enorme, descobrir na hora se existe ou não sessão para o filme que você quer ver... Hoje, com um celular você consegue fazer tudo: desde a avaliação da disponibilidade até a compra efetiva. E nem precisa imprimir o ingresso, apresentando o ticket na tela do celular você entra no cinema. Talvez não tenha aí na sua cidade, mas aqui em SP eu uso muito esse esquema.

Aos poucos, essas facilidades existentes em algumas áreas começam a criar um sentimento de dúvida: será que em outros ramos (sem ser o cinema), é possível criar sistemas que simplifiquem processos complexos? Esse tipo de dúvida ronda o mundo corporativo e de softwares empresariais diariamente.

Bem, sei que falar de apps corporativos para celulares, tablets e web é chover no molhado, mas muito pouco disso ainda é realidade no mundo SAP. Primeiro que o ABAPer médio mal fica sabendo dessas tecnologias de UI modernas, ele mal vê um Webdynpro na vida fora do GRC (me conta 10 telas de Webdynpro que você conhecer sem ser do GRC, valendooooo). Segundo que o caso mais levantado é sempre a aprovação de XYZ pelo celular. Perdi a conta de quantas vezes ouvi falar da empresa X ou Y que tem toda uma plataforma de mobilidade que serve para fazer aprovações e mais nada. Essa falta de inovação pode ser uma particularidade do mercado brasileiro, mas em parte acredito estar atrelada a licenças malucas e complexidade de explicar esse monte de siglas e variações para as empresas (dificultando, portanto, a venda).

Enfim, esse mundo é um caos, todos já sabem, **inclusive** a própria SAP. Engana-se você se pensa que a gigante alemã não sabe dos problemas de usabilidade de seus produtos. É muito fácil achar informação na web sobre SAP e UX, com [sites](#) dedicados ao tema e até [cursos](#) para explicar as novas possibilidades. Como dizia minha mãe: “quando você está indo, e/les já estão voltando”.

Pensando na nossa carreira como ABAPers, precisamos entender qual parte dessas novas coisas relacionadas a UI e experiência do usuário podem ter um impacto em nossas vidas de debuggadores. Particularmente, acredito que o SAP Netweaver Gateway é um dos produtos que têm mais chances de ver a luz do dia nas suas mãos digitadoras de código. Calma que eu explico.

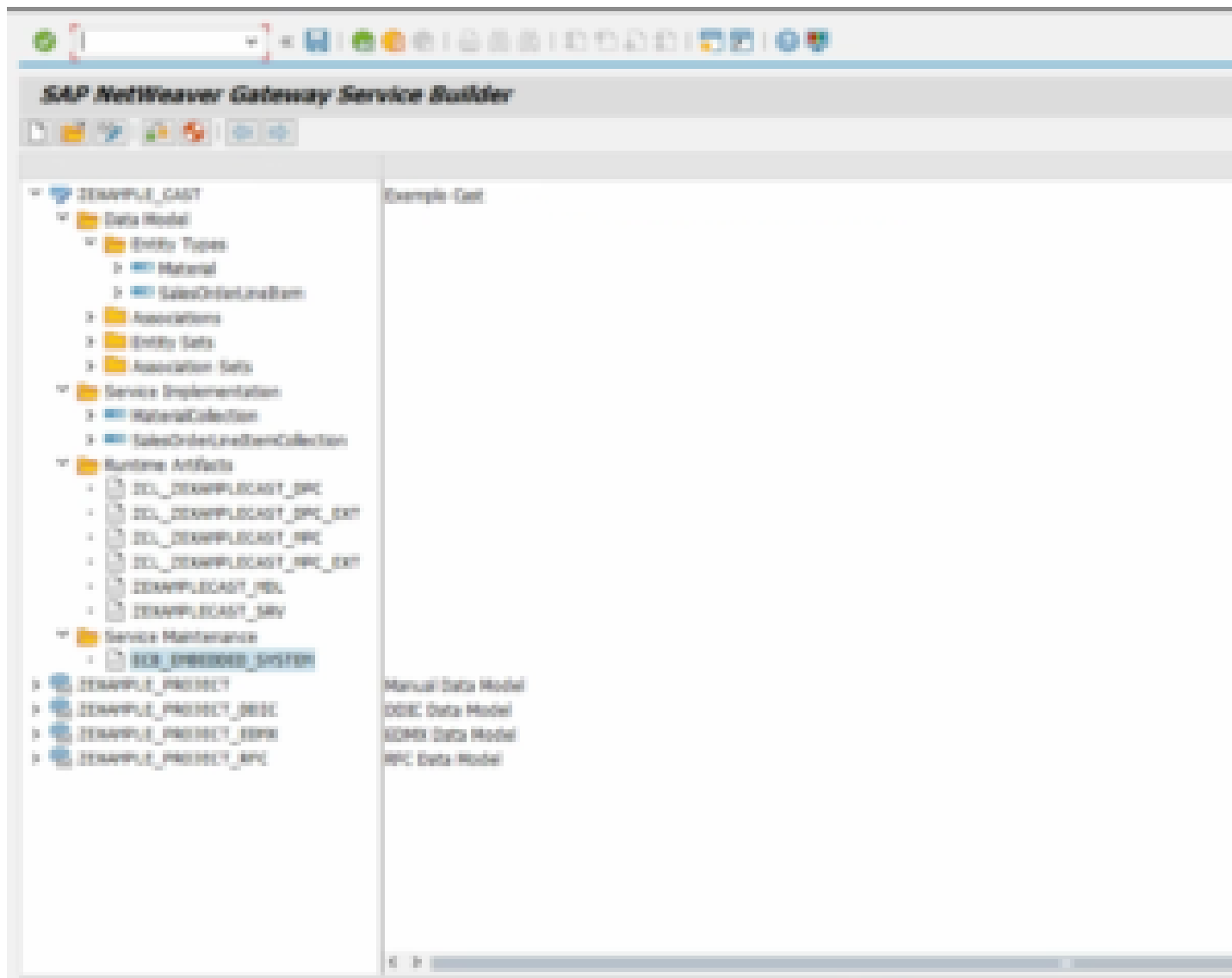
---

## “Me consuma”, diz o OData

Depois de toda essa coisa louca que eu falei de Interfaces, pense no esforço necessário para adaptar todas as telas e códigos do ERP (standards e customizadas) para tecnologias de UI modernas. Parece um trabalho insano, não é? Provavelmente, quando o trabalho estivesse concluído (se é que seja possível terminá-lo), as novas telas já estariam defasadas, dada a quantidade monstruosa de tempo necessário para fazer tal adaptação.

Não seria mais fácil, então, prover algum meio de aplicações externas acessarem os dados do ERP de uma forma consistente e gerenciável, desenvolvendo a UI em **outras** tecnologias? Dessa forma a SAP quebra as amarras das telas do seu sistema: o cliente pode consumir dados do ERP a partir de qualquer tipo de aplicação que ele quiser (e sonhar), criando uma UI específica para seus propósitos. É exatamente para tratar esse tipo de cenário que foi criado o *SAP Netweaver Gateway*.

Através do Gateway, desenvolvedores ABAP podem modelar um serviço [ODATA](#) (Open Data Protocol) para ser consumindo por alguma fonte externa. Essa fonte externa pode ser absolutamente qualquer sistema ou tecnologia que consiga consumir um serviço OData, seja ele um aplicativo web, um aplicativo mobile, um aplicativo desktop, uma máquina de café... Você cria esses serviços por uma transação específica, a SEGW:



Para entender o funcionamento completo do Gateway, é preciso estudar um pouco mais do protocolo ODATA e da arquitetura REST, mas isso sai do escopo deste singelo post. O **importante** é você saber que existe uma ferramenta da SAP que permite que um sistema externo Consulte, Atualize, Crie ou Apague dados do ERP, executando código ABAP. Nesse cenário, o ABAPer terá a função de desenvolvedor *backend*, tratando os dados e criando lógicas consistentes para que o aplicativo externo possa consumir a informação já trabalhada. Se você odeia desenhar telas e não entende nada de desenvolvimento *frontend*, essa parada de desenvolvedor *backend* pode ser um caminho para o seu futuro.

**Muitas siglas? Muitos nomes novos? Muita animação e confusão?** Vamos ser mais práticos.

Quero fazer uma tela muito louca para buscar dados de um material e preciso de algo do lado ABAP que retorne os dados de alguma forma, para que a minha aplicação web possa exibí-los em um quadrado que dá piruetas. Supondo que eu tenha criado um serviço OData com o Gateway, a app web vai lá e acessa o serviço OData por uma URL (a partir de um nó da SICF gerado pelo Gateway):



Que dispare um método de uma Classe do lado ABAP (também gerada pelo Gateway). Nesse método, você vai lá e faz seus SELECTs, BAPIs e marretadas para retornar os dados do Material. Isso então é formatado em um XML e retornado para o sistema web que chamou o serviço:

```

<?xml version='1.0' encoding='utf-8'>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices" xml:base="http://...:50000/sap/opu/odata/sap/ZEXAMFLECAST_SRV/">
  <id>
    http://...:50000/sap/opu/odata/sap/ZEXAMFLECAST_SRV/MaterialCollection
  </id>
  <title type="text">MaterialCollection</title>
  <updated>2014-07-28T14:38:40Z</updated>
  <author>
    <name/>
  </author>
  <link href="MaterialCollection" rel="self" title="MaterialCollection"/>
  <entry>
    <id>
      http://...:50000/sap/opu/odata/sap/ZEXAMFLECAST_SRV/MaterialCollection('Mat1')
    </id>
    <title type="text">MaterialCollection('Mat1')</title>
    <updated>2014-07-28T14:38:40Z</updated>
    <category term="ZEXAMFLECAST_SRV.Material" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme/">
    </category>
    <link href="MaterialCollection('Mat1')" rel="self" title="Material"/>
    <link href="MaterialCollection('Mat1')/SalesOrderLineItem" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/SalesOrderLineItem"
type="application/atom+xml;type=entry" title="SalesOrderLineItem"/>
    <link href="MaterialCollection('Mat1')/$links/SalesOrderLineItem"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/relatedlinks/SalesOrderLineItem" type="application/xml" title="SalesOrderLineItem"/>
    <content type="application/xml">
      <m:properties xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices">
        <d:MaterialID>Mat1</d:MaterialID>
        <d:CreationDate>01012014</d:CreationDate>
      </m:properties>
    </content>
  </entry>
  <entry>
    <id>
      http://...:50000/sap/opu/odata/sap/ZEXAMFLECAST_SRV/MaterialCollection('Mat2')
    </id>
    <title type="text">MaterialCollection('Mat2')</title>
    <updated>2014-07-28T14:38:40Z</updated>
    <category term="ZEXAMFLECAST_SRV.Material" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme/">
    </category>
    <link href="MaterialCollection('Mat2')" rel="self" title="Material"/>
    <link href="MaterialCollection('Mat2')/SalesOrderLineItem" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/SalesOrderLineItem"
type="application/atom+xml;type=entry" title="SalesOrderLineItem"/>
    <link href="MaterialCollection('Mat2')/$links/SalesOrderLineItem"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/relatedlinks/SalesOrderLineItem" type="application/xml" title="SalesOrderLineItem"/>
    <content type="application/xml">
      <m:properties xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices">
        <d:MaterialID>Mat2</d:MaterialID>
        <d:CreationDate>01012014</d:CreationDate>
      </m:properties>
    </content>
  </entry>
</feed>

```

Daí para frente, a aplicação que **consumiu** o serviço faz o que ela quiser.

Não precisa ser só uma busca: pode ser uma ação de deletar, criar, alterar... Esse monte de informações e tags que você está vendo aí em cima é um XML que segue um padrão hierárquico definido por você na hora de criar o serviço OData (prefere outros formatos, tipo o JSON? Também dá!). Aí no meio desse print tem as informações dos materiais e também das ordens de vendas ligadas ao material (caça o milho aê! 😊).

No Gateway, através da transação SEGW, você modela o serviço OData, definido a estrutura do serviço (dados de entrada/saída, associações, etc, etc) e ele cria para você as classes, entradas na SICF e todos os paranauês que são necessários para que alguém possa consumir o serviço. Tem ainda um monte de logs e variações e opções e usos e telas e tudo que você esperaria de algo da SAP.

Esse negócio é um mundo à parte. A explicação cobre um cenário muito básico, só para que você entenda do que se trata. Tem tanta coisa que precisa de um livro para explorar todas as possibilidades (tipo [esse](#) aqui).

## Mas ué, não é tipo o PI isso aí?

Esse conceito básico do Gateway é alvo de muita confusão com outro produto da SAP: o PI, *middleware* para gerenciamento de interfaces.

Todas as features do Gateway focam em prover os dados do ERP de forma que possam ser consumidos e/ou manipulados por aplicação externas que envolvam ações do usuário. Já o PI tem uma abrangência **muito** maior, atuando como um ponto único para integrações de qualquer natureza entre o SAP e sistemas externos. Apesar de parecerem similares, o posicionamento de cada ferramenta é **bem** diferente dentro do ecossistema.

## "E por que você acha que isso vai dar jogo?"

O SAP Netweaver Gateway é novo (criado em meados de 2011), portanto a aderência ainda não é ampla. Porém, veja só que bacana, a partir da versão 7.4 o Gateway vem embutido por **padrão** no pacote do ABAP. Não precisa de licença, ajoelhar no milho, benção do CIO, nada disso. É só ir lá, usar e fim.

Mesmo se você não estiver na 7.4, você ainda pode instalar o Gateway de forma separada, através de um add-on. [Este link](#) contém a disponibilidade por versões.

Ora, se a SAP criou a plataforma para prover dados para serem consumidos externamente, o que a impede de criar **seus próprios** aplicativos para exibir essas informações? Ela **já criou** pequeno mancebo, é um tal de SAP Fiori (que pretendo explicar num post futuro). Esses caras não dão ponto sem nó, é ou não é?

Termino assim esta singela explicação do Gateway. Espero que você tenha conseguido absorver pelo menos a idéia de existir uma coisa mágica que provê dados do ERP para serem consumidos, envolvendo programação ABAP. Agora volta lá para debuggar que daqui uns 3 ou 4 anos você usa isso (eu *nunca* disse que ia pegar rápido 😊).

Abraços a todos que já consumiram coisas bem piores que um serviço OData.

---

## Comentários

**Daniela Bender** — 19/10/2018 15:11

Show sua explicação! Nunca vi alguém tornar tão claro algo que todo mundo complica. Parabéns!

**Igor Muntoreanu** — 30/05/2016 13:37

Parabéns! Muito bem explicado! Abraço!

**Enio** — 25/04/2016 11:23

ótima explicação. Parabéns cara!

**Wagner Duarte** — 12/05/2015 10:17

Bacana o Post Mauricio Cruz.

Gostaria de discutir algumas questões a respeito. Através de RFC você já pode gerar um Webservice a ser consumido por qualquer legado usando protocolo SOAP e XML. Você também pode usar Abap proxy para gerar um webservice, ou pode consumir um serviço no SAP através dele. Não entendi muito bem a novidade do Gateway.

**Mauricio Cruz** — 13/05/2015 09:13

A grande novidade é o uso do protocolo OData com suporte a REST. Entretanto concordo que, dependendo do caso, passa ser somente mais uma alternativa. Se for para uma UI (que é exatamente o foco do Gateway), trabalhar com um serviço RESTful é muito mais fácil e intuitivo, sem falar a questão da performance (há um certo debate caloroso na net sobre performance de SOAP ser pior que a de um OData, o que tem impacto direto para apps que requerem muita interação do usuário).

Para entender melhor, vale a pena ler esse link aqui: <http://www.developingthefuture.net/web-services-overview/> , que faz uma comparação bem bacana de REST x SOAP x OData.

Abs!

**Gabriel** — 23/06/2015 13:24

Mauricio, tudo bem?

E para o caso da nova versão do SAP PI/PO 7.4, que já possui suporte a OData services, OData adapter, REST adapter, etc... Eu fui no SAP Forum 2015, e alguns disseram que talvez o SAP PO venha a arrancar o mercado do gateway... Gostaria de Saber sua opinião. 😊

Muito obrigado, adorei o post!

Abs

**Mauricio Cruz** — 06/07/2015 09:22

Gabriel, tudo bem?

Acho pouco provável que o SAP PO venha a tomar tanto espaço assim do gateway, já que a própria SAP faz o deploy dos apps Fiori usando serviços OData gerados pelo Gateway.

Mas sabe como é, tudo pode mudar 😊

Abs!

**Flavio** — 25/03/2015 16:20

É verdade, concordo com o Custódio, já ouço falar de GW desde de antes de 2012, mas não dei muita bola, afinal iria demorar para eu usar. Estamos em 2015 e ainda não consigo usar aqui por questão de "não ser permitido pela política da firma". Continuemos com o debugger, pelo menos é o novo 😊

Abraços!

**Thiago Alves** — 06/03/2015 09:30

Ótimo post. Comecei a abrir a mente para esse tipo de coisa agora que estou tendo que desenvolver uma BSP... MVC, Bootstrap, Javascript, e várias coisas que antes eu só ouvia falar, estou tendo que aprender e aplicar nesse desenvolvimento... Com certeza vou correr atrás e tentar aprender mais, acho que tudo indica que quem não se adaptar à esse "novo" mundo, logo terá problemas...

**Anderson Miazato** — 20/02/2015 08:03

Eu sempre olho o blog, mas nunca comento kkk.

É fácil ficar perdido no meio de tanta coisa "aparecendo" (algumas nem são tão novas assim, só demora um pouco até serem reconhecidas, como você disse). Através do blog dá pra ter uma boa ideia do que são as tecnologias que podem virar tendência daqui um tempo, e aí fica mais fácil pra correr atrás e estudar.

Valeu pelo post e pelo blog, Mauricio!

**Custodio** — 15/02/2015 20:12

Fala Mauricio,

Post marotao, como sempre. So duas nao exatamente correcoes porque nao tenho as respostas certas, mas sao mais chatice minha mesmo 😊

1 – Gateway nao foi “criado” em 2012. Pode ate ter sido GA em 2012, mas com certeza ja estava disponivel antes disso, pois entrei em um cliente em jun/2011 e foi considerado usar GW, mas um dos motivos de nao usar foi o lance da licenca. O que nos leva ao 2nd ponto...

2 – GW nao vem embutido no pacotao nem de software nem de licenca “a partir da 7.4” . No meu ultimo cliente estao na 7.31 e ja tem GW “de gratis e esta sendo bastante usado.

Abraco,  
Custodio

**Mauricio Cruz** — 13/05/2015 09:15

Subiram o kernel aqui do ABAP na minha empresa para o 7.4 e o gateway veio junto no pacote, sem precisar instalar nenhum addon adicional.

Ah e eu falei que na 7.4 ele já vem incluso e não que ele não está disponível em versões anteriores. Arrumei o texto para esse ponto ficar mais claro.

Valeu, abs!

**Sérgio Paes** — 11/02/2015 08:59

Bem legal o post Mauricio, parabéns!

Eu acompanho o teu site desde o ano passado, quando comecei a estudar SAP e tem me ajudado um monte mesmo!

Eu sou consultor em JD Edwards e Mastersaf, porém, agora venho me envolvendo em projetos de integração SAP-Mastersaf-NFe. E não tem como não ficar impressionado com esse mundo SAP.

Vi o comentário do Gabriel ali acima e fui verificar no Open Sap e iniciei um curso, creio que seja o mesmo que ele citou (<https://open.sap.com/courses>) – Introduction to SAP Fiori UX.

Grande abraço e sucesso!

**Mauricio Cruz** — 11/02/2015 10:37

Valeu Sérgio! Fique sempre ligado no openSAP, os cursos têm bastante qualidade e alguns só podem ser feitos em datas específicas.

Abs!

**Ricardo** — 28/01/2015 16:13

Muito bom o conteúdo do site, parabéns!

**Gabriel Graça** — 20/01/2015 10:37

Muito legal o post Maurício! Eu vi esse netweaver gateway em um curso de SMP3.0, e achei bem interessante. Atualmente estou estudando o FIORI no open.sap.com pra me atualizar um pouco, legal que você vai começar a colocar posts dessas novas siglas que saíram pq as vezes eu fico meio perdido.  
Abs!

**Mauricio Cruz** — 23/01/2015 10:19

Valeu pelo comentário Gabriel!