

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DO RIO GRANDE DO SUL
CAMPUS RESTINGA**

**MEMORIAL PROFISSIONAL
ATUAÇÃO COMO DESENVOLVEDOR JÚNIOR, 2022 A 2024**

PEDRO ARTHUR PUGNES SARATE

**PORTO ALEGRE
2025**

PEDRO ARTHUR PUGNES SARATE

**MEMORIAL PROFISSIONAL
ATUAÇÃO COMO DESENVOLVEDOR JÚNIOR, 2022 A 2024**

Trabalho de Conclusão de Curso apresentado junto ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas Superior do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul - Campus Restinga, como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador:

Prof. Dr. Gilberto João Pavani.

PORTO ALEGRE

2025

PEDRO ARTHUR PUGNES SARATE

**MEMORIAL PROFISSIONAL
ATUAÇÃO COMO DESENVOLVEDOR JÚNIOR, 2022 A 2024**

Trabalho de Conclusão de Curso apresentado junto ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas Superior do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul - Campus Restinga, como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador:

Prof. Dr. Gilberto João Pavani.

Aprovada em Fevereiro de 2025.

Prof. Dr. Gilberto João Pavani - Orientador

Prof. André Forlin Dosciati - IFRS/Campus Restinga

Prof. Fausto Armando Bischoff - IFRS/Campus Restinga

“Aquilo que não puderes controlar, não ordenes.”

Sócrates

RESUMO

Durante minha atuação como desenvolvedor júnior na Neogrid, entre 2022 e 2024, participei de atividades voltadas à transformação e conectividade de dados, contribuindo para a integração entre clientes e a cadeia de suprimentos. A Neogrid, empresa consolidada no mercado, fundada em 1999, destaca-se no setor de gestão de cadeia de suprimentos, atuando em parceria com grandes marcas globais. Meu trabalho focou na integração de dados via plataforma WebMethods, ferramenta que possibilita a conversão e transmissão de arquivos em diversos formatos, como JSON e XML, o que facilita a comunicação eficiente entre sistemas diferentes. Minhas principais atividades incluíram processos de transformação de dados, garantindo que arquivos entre parceiros comerciais fossem convertidos de forma a atender os formatos específicos de cada cliente, e a conectividade de dados, viabilizando a troca segura de informações entre Neogrid, clientes e parceiros. Os conhecimentos adquiridos no curso foram fundamentais para o desempenho de meu trabalho, pois construíram uma base sólida, proporcionando maior celeridade no aprendizado quando era necessário desempenhar uma nova atividade. O curso também apoiou em momentos onde era necessária a resolução de problemas técnicos.

Palavras-chave: Neogrid, integração de dados, WebMethods, transformação de arquivos, conectividade, cadeia de suprimentos.

ABSTRACT

During my time as a junior developer at Neogrid from 2022 to 2024, I engaged in activities focused on data transformation and connectivity, contributing to the integration between clients and the supply chain. Neogrid, a well-established company founded in 1999, stands out in the supply chain management sector, working alongside major global brands. My work focused on data integration via the WebMethods platform, a tool that enables the conversion and transmission of files in various formats, such as JSON and XML, facilitating efficient communication between different systems. My main activities included data transformation processes, ensuring that files between commercial partners were converted to meet the specific formats required by each client, and data connectivity, enabling the secure exchange of information between Neogrid, clients, and partners. The knowledge acquired throughout my course was essential to my job performance, as it provided a solid foundation, allowing me to learn more quickly when it was necessary to perform a new task. The course also supported me in times when technical problem-solving was needed.

Keywords: Neogrid, data integration, WebMethods, file transformation, connectivity, supply chain.

LISTA DE FIGURAS

Figura 1 – Pacote Parceiro	21
Figura 2 – Esqueleto de serviço de exportação	21
Figura 3 – Execução de serviço	22
Figura 4 – Preenchendo variáveis do serviço mercador.migracao_essencial:migracao_1	22
Figura 5 – Pasta records após execução do serviço mercador.migracao_essencial:migracao_1	22
Figura 6 – Esqueleto do serviço principal	23
Figura 7 – SEQUENCE getRecord	23
Figura 8 – SEQUENCE getProperties	23
Figura 9 – Mapeamentos serviço	24
Figura 10 – SEQUENCE Initialize serviço principal	24
Figura 11 – SEQUENCE mapeamentos do serviço	24
Figura 12 – SEQUENCE adjust do serviço	25
Figura 13 – Configurando serviço pub.flatFile:convertToString	25
Figura 14 – Tipos de Initialize	26
Figura 15 – Inicializando campos no Initialize único	27
Figura 16 – Criando Initialize para array	28
Figura 17 – Copiando documentList para utilizar no Initialize	29
Figura 18 – Alterando ocorrência do registro para Document	29
Figura 19 – Inicializando campos no Initialize array	30
Figura 20 – Usando mapeando o initialize array para o registro dentro do LOOP	31
Figura 21 – Configurando o Input/Output dos mapeamentos	32
Figura 22 – Configurando o Adjust	32
Figura 23 – Configurando o Input/Output do Adjust	33
Figura 24 – Configurando LOOPS no Adjust	34
Figura 25 – Serviço principal após adição dos outros serviços desenvolvidos . .	34
Figura 26 – Initialize para campos únicos	35
Figura 27 – Configurando pipeline do mapping único	35
Figura 28 – Configurando pipeline do Adjust	35
Figura 29 – Configurando serviço convertToString	36

Figura 30 – Configurando serviço convertToValues	36
Figura 31 – Pacote e pastas	38
Figura 32 – Templates de SFTP	38
Figura 33 – Adaptando nomenclatura dos serviços	39
Figura 34 – Configurando alias do servidor e usuário	39
Figura 35 – Criando alias do servidor	40
Figura 36 – Criando alias do usuário	41
Figura 37 – Testando conexão SFTP	41
Figura 38 – Preenchendo variáveis do serviço getSFTPCustomerProps	42

LISTA DE ABREVIATURAS E SIGLAS

DAD	Demand Activation Distribution
EDI	Electronic Data Interchange
DRP	Demand Replenishment Planning
DAR	Demand Activation Retail
B2B	Business-to-Business
SaaS	Software as a Service
CSV	Comma-separated values
JSON	JavaScript Object Notation

SUMÁRIO

1	INTRODUÇÃO	14
2	CONTEXTUALIZAÇÃO DA EMPRESA	16
3	ATIVIDADES DESENVOLVIDAS & PROJETOS REALIZADOS	18
3.1	Contextualização da empresa fornecedora da ferramenta que utilizo para trabalho	18
3.2	Contextualizando o produto	19
3.3	Atividades desenvolvidas	19
3.3.1	Transformação	19
3.3.2	Passo a Passo	20
3.3.3	Conectividade	37
3.3.4	Passo a Passo	37
3.4	Projetos	42
3.4.1	EDP	42
3.4.2	Berlanda	43
4	RELAÇÃO DAS ATIVIDADES COM OS COMPONENTES CURRICULARES E APRENDIZADO NO CURSO	44
5	CONCLUSÃO	46
	REFERÊNCIAS	47

1 INTRODUÇÃO

Este é um relatório sobre minha experiência profissional na empresa Neogrid, a qual tem uma trajetória notável desde a sua fundação em 1999. Durante minha permanência na Neogrid, atuei como desenvolvedor Júnior, abrangendo o período de janeiro de 2022 até janeiro de 2024, abordado neste relatório.

A Neogrid, desde o seu início, se destacou no mercado por sua missão dedicada a oferecer soluções de gestão de cadeia de suprimentos (Supply Chain Management) de alta qualidade e eficácia. No decorrer do meu tempo na empresa, integrei o time de "Ingestão de Dados", um setor fundamental para o funcionamento da empresa, pois sem este time a Neogrid não pode receber nem transmitir dados para seus processos de enriquecimento, tratamento e auditoria de dados.

Minha função consistia em desempenhar um papel vital, o desenvolvimento de serviços relacionados à tradução de arquivos e conectividade entre a Neogrid e o cliente, utilizando a plataforma de software WebMethods, desenvolvida pela empresa Software AG. Através dessas ferramentas, fui responsável por facilitar a comunicação e a integração de dados para o funcionamento da cadeia de suprimentos dos clientes.

Neste relatório, vou compartilhar as experiências e os desafios que enfrentei durante meu período como desenvolvedor Júnior na Neogrid, bem como destacar as contribuições que pude trazer para a equipe e para a empresa como um todo. Meu objetivo é fornecer uma visão abrangente do meu trabalho e do ambiente profissional na Neogrid, destacando o meu papel no apoio a empresa no domínio de Supply Chain Management. Participei de alguns projetos que trouxeram uma receita considerável para a Neogrid, como: projeto Berlanda e projeto EDP.

Esses dois projetos tiveram o objetivo de desenvolver uma conexão entre as bases de dados dos clientes e os produtos Neogrid contratados. Cada projeto foi específico para um cliente, neles o time de Ingestão de Dados precisou desenvolver

formas customizadas para cada cliente de busca de dados em seu sistema e transformação para entregar no produto Neogrid desejado. Nestes projetos, desempenhei papéis-chave para seu desenvolvimento e entrega.

2 CONTEXTUALIZAÇÃO DA EMPRESA

A empresa é denominada NEOGRID INFORMATICA LTDA, operando sob o nome fantasia NEOGRID, possui cerca de 900 colaboradores espalhados pelo Brasil, EUA e Holanda. Seu CNPJ é 05.794.609/0001-01. O endereço da matriz está localizado na Avenida Santos Dumont, número 935, no quarto andar, na região de SANTO ANTONIO, na cidade de JOINVILLE, estado de Santa Catarina (SC), com o CEP 89218-105. A Neogrid tem como atividades de negócio: desenvolvimento e licenciamento de programas de computador customizáveis, consultoria em tecnologia da informação, suporte técnico, manutenção e outros serviços em tecnologia da informação e holdings de instituições não-financeiras. Ela possui mais de 40 mil clientes espalhados pelos 4 continentes, incluindo algumas marcas com presença global como Bunge, Samsung, Apple, Ambev, P&G, Mondelez, Unilever e Nestlé. A organização possui um portfólio de mais de 15 produtos ofertados para os diferentes níveis da cadeia de suprimentos que são: indústria, varejo e distribuidores. Aqui vai uma breve descrição de 4 produtos:

- DAD - A indústria tem acesso aos dados diários de performance dos distribuidores e trabalha junto com eles para aumentar as vendas.
- EDI - Com este produto é possível automatizar a troca de pedidos, cobranças e documentos logísticos com os parceiros de negócio.
- DRP - Torna o processo de abastecimento e distribuição mais rentável com inteligência artificial, evitando faltas ou excessos de estoque.
- DAR - Utilizando este produto conseguimos acompanhar a performance diária e o estoque dos produtos em cada ponto de venda.

O regime de trabalho começou sendo remoto devido a pandemia, portanto a comunicação com minha equipe sempre ocorreu via Microsoft Teams tanto para chat quanto para reuniões online. Posteriormente, foi colocado um aditivo contratual onde

alterava o regime para híbrido, então tive alguns encontros presenciais no escritório com quase todos os meus colegas de equipe, porque alguns deles residem em outros estados. Tínhamos uma reunião diária onde compartilhávamos impedimentos que surgiam nas demandas do dia a dia e pedíamos apoio à equipe ou ao tech leader para resolução desses travamentos.

3 ATIVIDADES DESENVOLVIDAS & PROJETOS REALIZADOS

3.1 Contextualização da empresa fornecedora da ferramenta que utilizo para trabalho

A webMethods, fundada em 1996 por Phillip Merrick e Caren Merrick, destacou-se como uma empresa de software empresarial focada na integração de aplicativos, processos de negócios e parceiros B2B. Especializada em utilizar padrões da web, como HTTP e XML, a empresa proporcionou a comunicação em tempo real entre aplicativos de software.

Ao abrir o capital na NASDAQ em 2000, a webMethods realizou o IPO de software mais bem-sucedido até então, evidenciando o interesse dos investidores e a valorização das ações. Em 2002, foi reconhecida pela Deloitte como a empresa de software que mais cresceu na América do Norte de 1998 a 2002.

Em 2007, a Software AG adquiriu a webMethods por US \$ 546 milhões, mantendo seu nome como marca principal. A empresa continuou inovando, lançando o webMethods versão 8.0 em 2009, integrado com outros produtos da Software AG. Em 2010, a divisão webMethods registrou receitas de mais de US \$ 668 milhões, contribuindo significativamente para o lucro líquido geral da empresa, que atingiu US \$ 292 milhões.

A webMethods foi pioneira no desenvolvimento de padrões para tecnologias de serviços web, participando do XML-RPC e desenvolvendo o WIDL (Web Interface Definition Language). Seu legado continua sendo reconhecido na indústria de software empresarial.

3.2 Contextualizando o produto

O produto/ferramenta webMethods, é uma plataforma de integração que oferece uma ampla gama de recursos para conectar aplicativos, processos de negócios e parceiros B2B. Sua versatilidade o torna aplicável em diversos cenários e setores, sendo uma solução abrangente para os desafios de integração enfrentados pelas organizações. Abaixo, destaco algumas das principais utilidades e cenários nos quais o webMethods pode ser empregado:

- **Integração de Aplicativos Empresariais:** o webMethods facilita a integração de diferentes sistemas e aplicativos dentro de uma organização. Ele permite a comunicação eficiente entre sistemas legados, aplicações locais e em nuvem.
- **Automatização de Processos de Negócios:** ele oferece recursos para automatizar processos de negócios, desde tarefas simples até fluxos de trabalho complexos. Isso resulta em maior eficiência, redução de erros e aceleração do ciclo de vida dos processos organizacionais.
- **Integração B2B (Business-to-Business):** No contexto de parcerias comerciais, o webMethods facilita a troca segura de informações entre diferentes empresas. Isso é vital em cadeias de suprimentos e transações comerciais.
- **Melhoria de Processos e Ativação de SOA (Arquitetura Orientada a Serviços):** O webMethods suporta a implementação de arquiteturas orientadas a serviços, promovendo a modularidade e reutilização de serviços. Isso é crucial para a modernização da TI e a adaptação às mudanças nas demandas de negócios.

Um aspecto fundamental é que a sua IDE webMethods Designer utiliza low-code para o desenvolvimento de serviços, por padrão temos inúmeros pequenos serviços para diversas operações como consultas SQL, pode efetuar leitura e escrita de arquivos locais ou remotos e enviar dados via requisições de um Webservice SOAP, REST, SFTP e AS2.

3.3 Atividades desenvolvidas

As atividades desenvolvidas por mim, consistiam em dois grupos: transformação e conectividade.

3.3.1 Transformação

As atividades essenciais de transformação consistiram na conversão de arquivos entre diferentes formatos, exemplificando, por meio da transformação de arquivos

JSON em XML e vice-versa. Essa prática tem como objetivo primário viabilizar a troca eficiente de arquivos e informações comerciais entre duas empresas que utilizam formatos específicos de dados. Um exemplo prático dessa necessidade é observado na empresa Kabum, que busca realizar pedidos de produtos da Apple, como iPads, iPhones e MacBooks. Contudo, a Kabum gerencia todos os seus documentos comerciais em formato CSV, enquanto a Apple utiliza exclusivamente arquivos JSON.

Diante dessa disparidade de formatos, a Neogrid surge como uma solução viável, oferecendo serviços de transformação para a realização dessa troca de informações. Essa prática está integrada ao produto EDI, assegurando a conversão eficaz e eficiente dos dados, mas também garantindo confiabilidade, visualização clara e rastreamento detalhado de cada informação.

Irei descrever abaixo como é feito o desenvolvimento de um serviço de transformação de exportação, ou seja, transformar de XML para qualquer tipo de formato de saída, neste exemplo será um Flat File.

Primeiro irei descrever brevemente alguns dos componentes lógicos que são utilizados na programação low-code da IDE do WebMethods.

- MAP: permite o mapeamento de dados de uma estrutura de entrada para outra. Pode ser usado para manipular, transformar ou configurar valores durante o fluxo de execução.
- BRANCH: é um elemento condicional que implementa lógica de decisão dentro do fluxo.
- LOOP: ele permite a repetição de ações sobre uma lista de itens (arrays, documentos, etc.).

3.3.2 Passo a Passo

1- Exportar o Schema e o Dicionário da plataforma de armazenamento de layouts.

Para iniciar, faz-se necessário exportar o Dicionário e o Schema da plataforma onde são armazenados os layout dos clientes fora do webMethods. Nome padrão para o Dicionário: dictionary_CLIENTE_DOCUMENTO Nome padrão para o Schema: schema_CLIENTE_DOCUMENTO

2- Criando hierarquia de pastas

Padrão de pastas: NomeEmpresa > mercador > webedi > NomeEmpresa > tipoDoc Se já existir um Pacote para empresa que o serviço será criado, é necessário apenas criar a pasta tipoDoc.

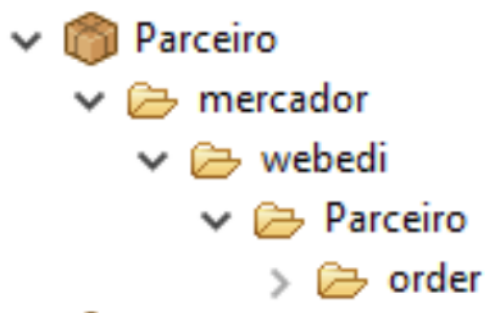


Figura 1 – Pacote Parceiro

3 - Copiar esqueleto do serviço de exportação

Localize o Pacote MercadorIntegracao > esqueletoGenericoServicosFF > ServicoExportacaoFF. Copie os seguintes arquivos e cole na pasta do serviço em questão:

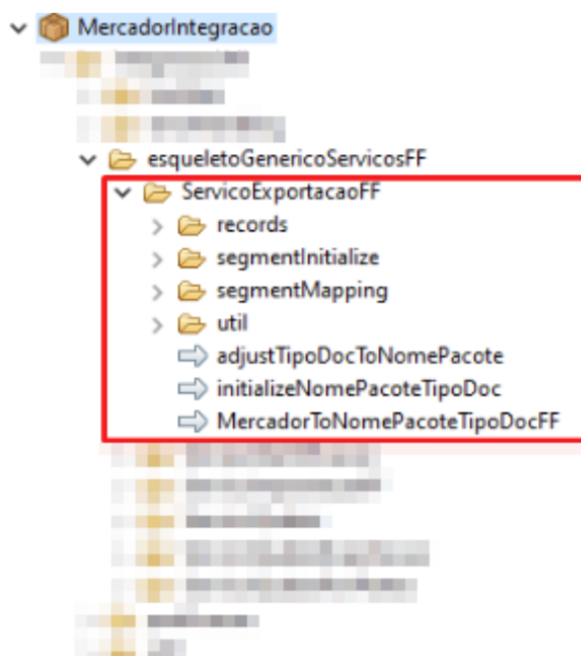


Figura 2 – Esqueleto de serviço de exportação

4 - Realizar a migração dos records

Após ter exportado o Dicionário e o Schema do serviço na plataforma, é

necessário gerar os Records. Usar o serviço mercador.migracao_essencial:migracao_1, clicar em Run > Run As > 1 Run Flow Service e preencher conforme a Figura 3:

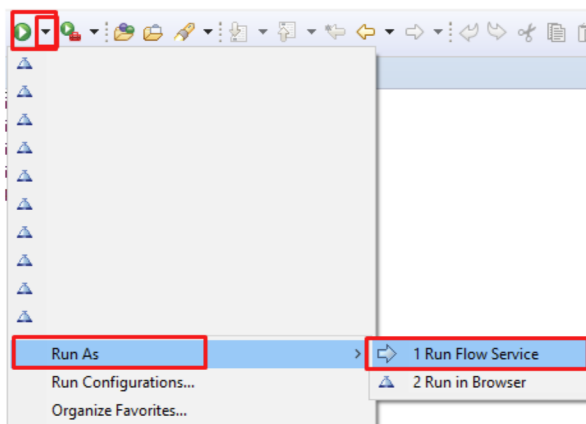


Figura 3 – Execução de serviço

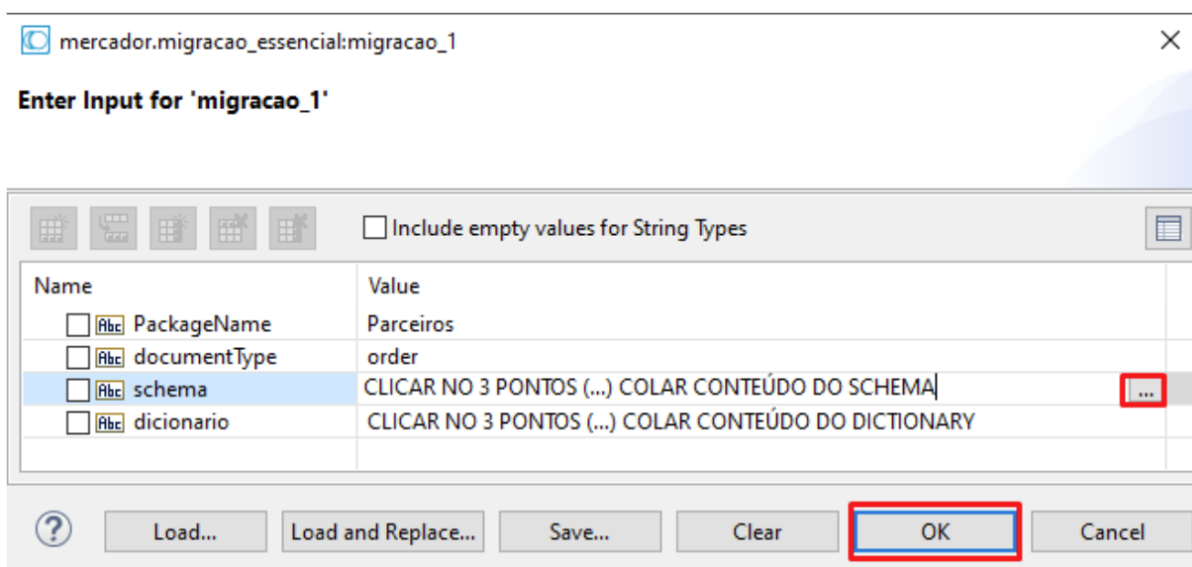


Figura 4 – Preenchendo variáveis do serviço mercador.migracao_essencial:migracao_1

A pasta records ficará como na Figura 5.

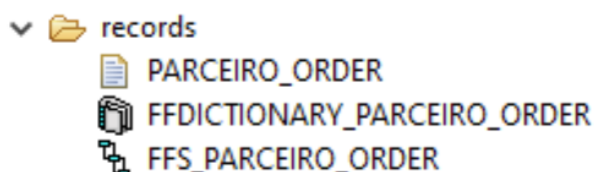


Figura 5 – Pasta records após execução do serviço mercador.migracao_essencial:migracao_1

5 - Configurar o Serviço Principal

O serviço principal (MercadorToNomePacoteTipoDoc) que foi copiado do esqueleto do serviço de Exportação terá a seguinte estrutura:

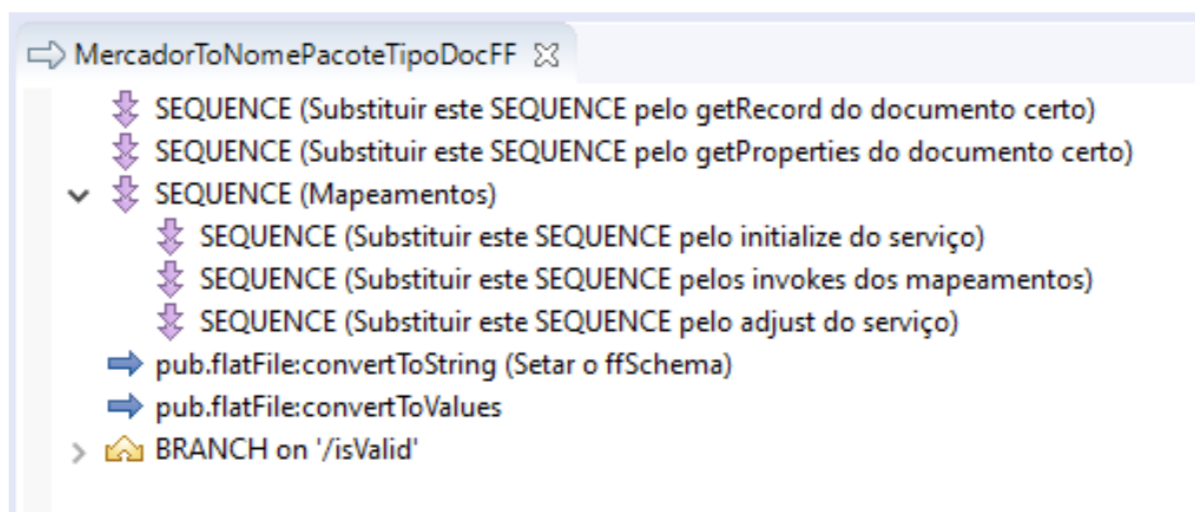


Figura 6 – Esqueleto do serviço principal

Podemos identificar que todos os SEQUENCE possuem uma breve descrição do que deve ser feito para realizar a configuração do serviço. Abaixo serão descritos todos os passos de como configurar cada SEQUENCE.

5.1 - SEQUENCE (Substituir o SEQUENCE pelo getRecord do tipo de documento certo)

SEQUENCE (Substituir este SEQUENCE pelo getRecord do documento certo)

Figura 7 – SEQUENCE getRecord

Este SEQUENCE deve ser substituído pelo arquivo de getRecord para o documento em questão, este getRecord se encontra no pacote MercadorRecords. O padrão é getMercadorDoctypeRecord.

5.2 - SEQUENCE (Substituir o SEQUENCE pelo getProperties do tipo de documento certo)

SEQUENCE (Substituir este SEQUENCE pelo getProperties do documento certo)

Figura 8 – SEQUENCE getProperties

O SEQUENCE deve ser substituído pelo arquivo de getProperties do Mercador para o documento em questão, este getProperties se encontra no pacote MercadorRecords. O padrão é getDoctypeProperties.

5.3 - SEQUENCE (Mapeamentos)

- ✓
 - ↓ SEQUENCE (Mapeamentos)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelo initialize do serviço)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelos invokes dos mapeamentos)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelo adjust do serviço)

Figura 9 – Mapeamentos serviço

Este SEQUENCE (Mapeamentos) não deve ser excluído para que o serviço fique organizado.

5.3.1 - SEQUENCE (Substituir o SEQUENCE pelo initialize do serviço)

- ✓
 - ↓ SEQUENCE (Mapeamentos)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelo initialize do serviço)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelos invokes dos mapeamentos)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelo adjust do serviço)

Figura 10 – SEQUENCE Initialize serviço principal

Este SEQUENCE deve ser substituído pelo arquivo de iniatialize do serviço que foi gerado a partir do esqueleto utilizado (initializeNomePacoteTipoDoc).

5.3.2 - SEQUENCE (Substituir o SEQUENCE pelos invokes dos mapeamentos criados)

- ✓
 - ↓ SEQUENCE (Mapeamentos)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelo initialize do serviço)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelos invokes dos mapeamentos)
 - ↓ SEQUENCE (Substituir este SEQUENCE pelo adjust do serviço)

Figura 11 – SEQUENCE mapeamentos do serviço

Este SEQUENCE deve conter os invokes dos mapeamentos. Os arquivos de mapeamento também foram gerados a partir do esqueleto e estão disponíveis dentro da pasta segmentMapping.

5.3.3 - SEQUENCE (Substituir este SEQUENCE pelo adjust do serviço)

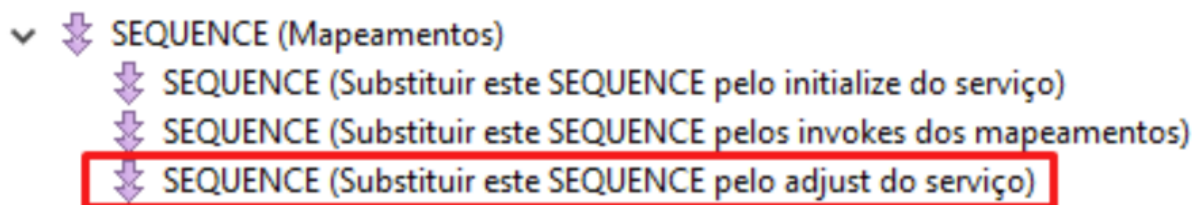


Figura 12 – SEQUENCE adjust do serviço

Este SEQUENCE deve ser substituído pelo arquivo de adjust do serviço que foi gerado a partir do esqueleto utilizado (adjustTipoDocToNomePacote).

5.4 - pub.flatFile:convertToString (Setar o ffSchema) Ainda no serviço principal, na janela Pipeline, é necessário setar o valor da variável ffSchema. Para isto, basta dar duplo-clique na variável ffSchema mostrada em Service In e definir o schema que é um dos arquivos gerados está na pasta records do pacote. O schema deverá ser definido com o seu caminho junto, um exemplo é `mercador.webedi.Parceiro.order.records:FFS_PARCEIRO_ORDER`.

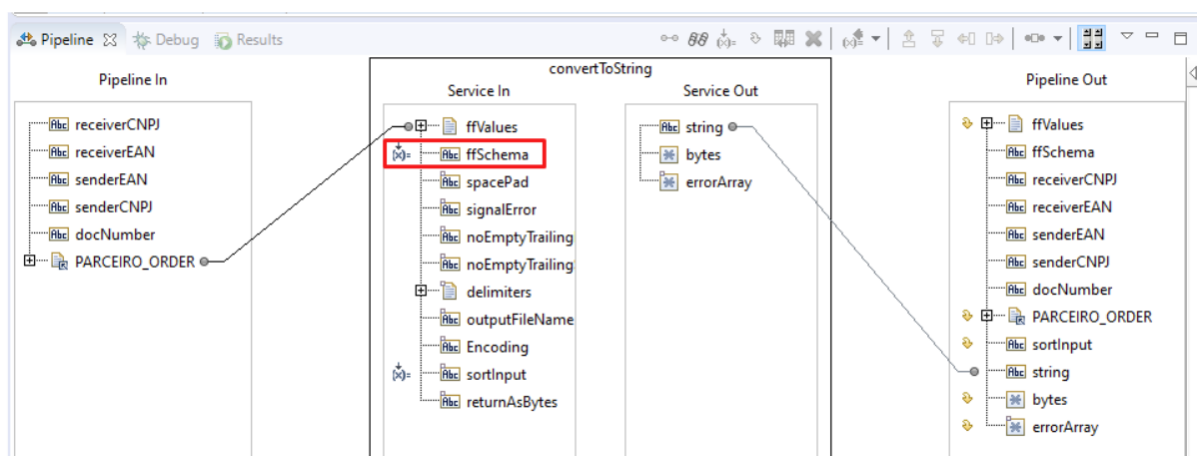


Figura 13 – Configurando serviço pub.flatFile:convertToString

6 - Configurando o serviço Inicialize

Este serviço serve para inicializar os registros que serão exportados no layout do cliente. Os inicializes são separados de acordo com a estrutura do registro, ou

seja, a inicialização de registros únicos é feita no serviço initializeNomePacoteTipoDoc, enquanto a inicialização de registros array é feita em outro(s) serviço(s) (um por registro array), estes estarão dentro da pasta segmentInitalize.

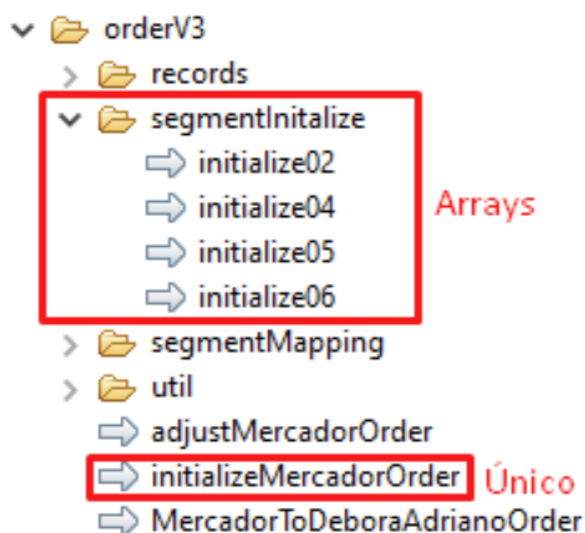


Figura 14 – Tipos de Initalize

6.1 - Inicialização de Registro Único

Os registros únicos são caracterizados por terem um número de ocorrência máxima igual a 1. Para verificar quais registros são únicos, basta verificar o campo Ocorrência Máxima na aba Registros da plataforma de layouts. Todos os campos do registro único devem ser inicializados com vazio, para isto deve clicar duas vezes no campo e clicar em OK, porém para os campos obrigatórios @record-id e @segment-id devem ser inicializados com o Identificador do Registro, para isto deve clicar duas vezes no campo, inserir o valor desejado e clicar em OK (caso não tenha o Identificador do Registro cadastrado no mapping basta clicar em OK). Veja como ficará o initialize de registros únicos:

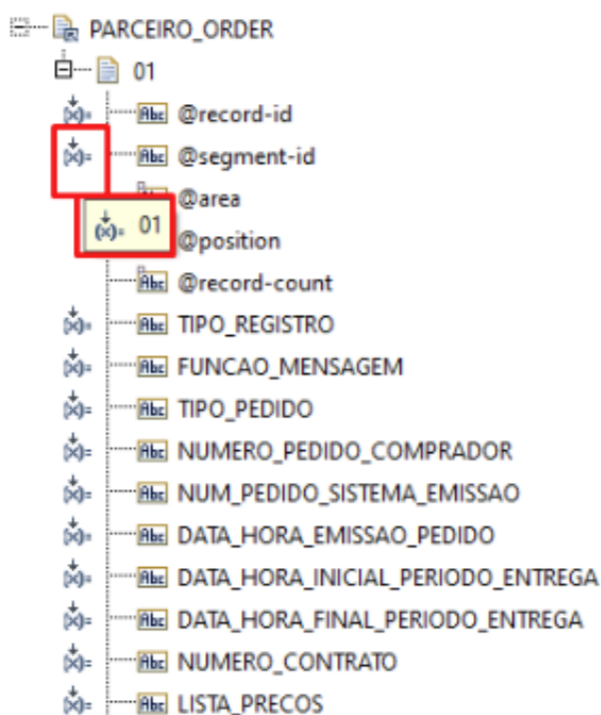


Figura 15 – Inicializando campos no Initialize único

6.2 - Inicialização de Registro Array

Para cada registro array deverá ser criado seu próprio Initialize dentro da pasta segmentInitialize. Para isto, clique com o botão direito na pasta segmentInitialize > New > Flow Service > Insira o nome em Element name seguindo o padrão InitializeNomeRegistro > Clique em Finish.

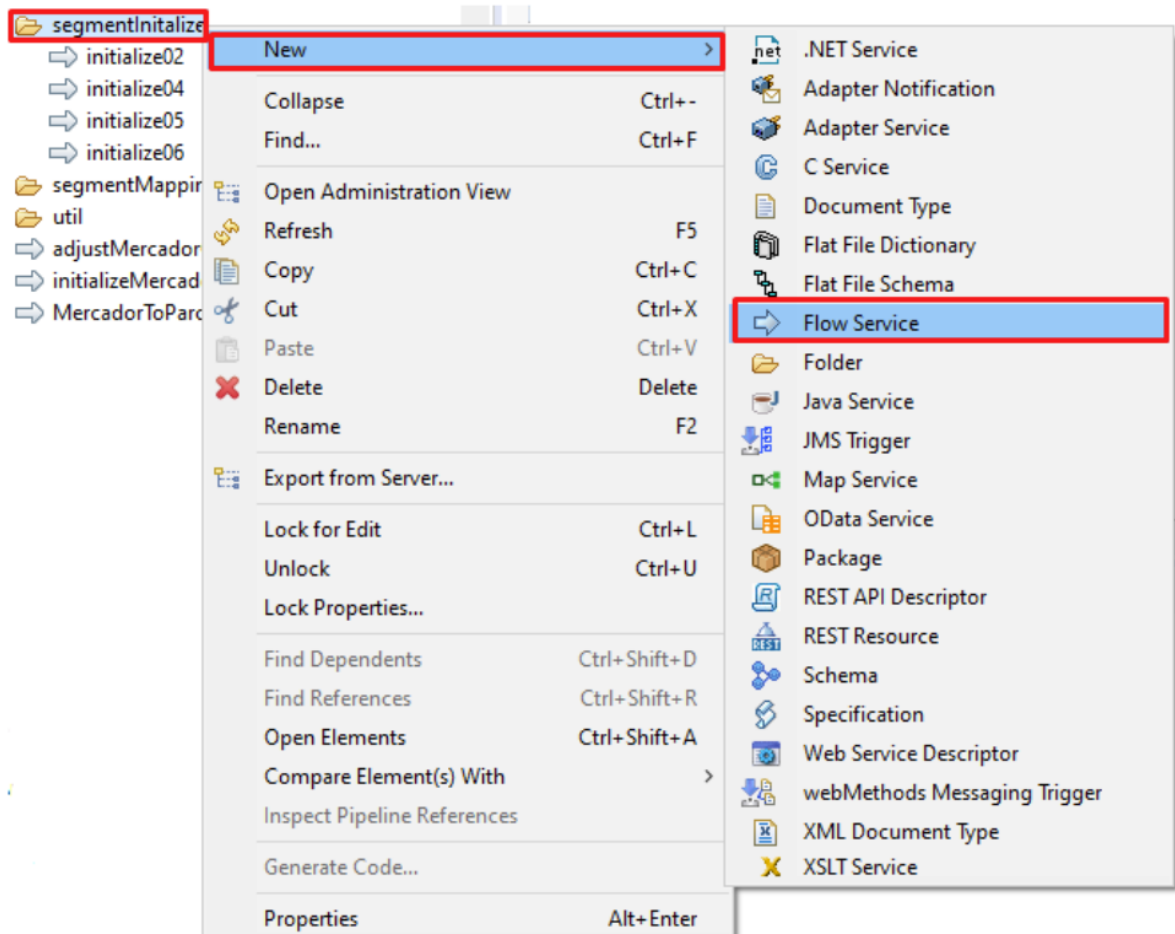


Figura 16 – Criando Initialize para array

Após a criação do initialize abra o Layout do serviço e copie o registro array que irá inicializar, após encontre a aba Input/Output do initialize criado, cole este registro no Output, clique no registro, vá em Properties e altere o Data type para Document. Conforme imagens abaixo:

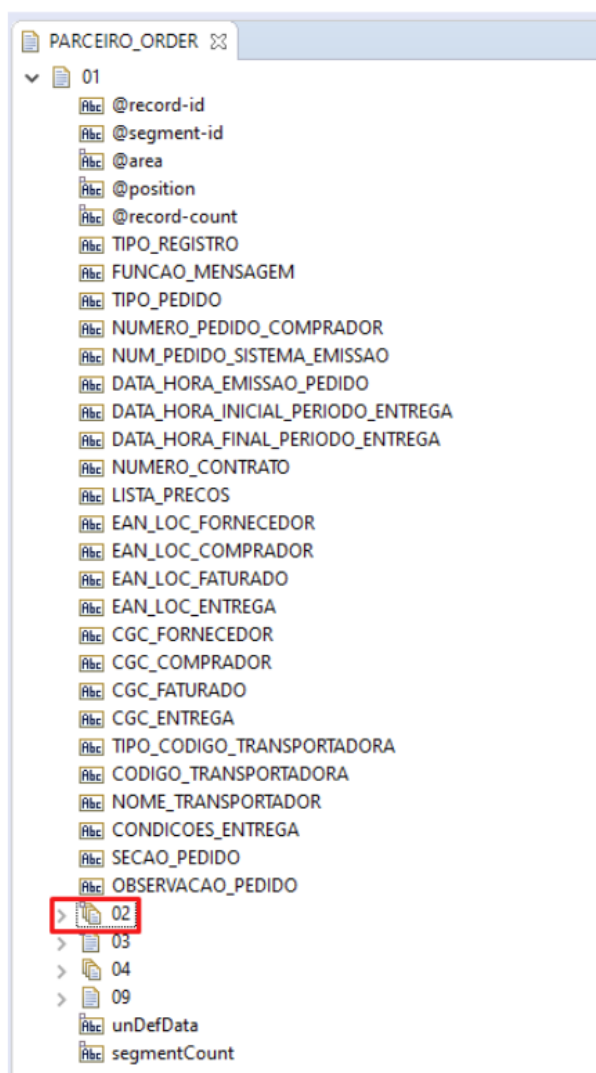


Figura 17 – Copiando documentList para utilizar no Initialize

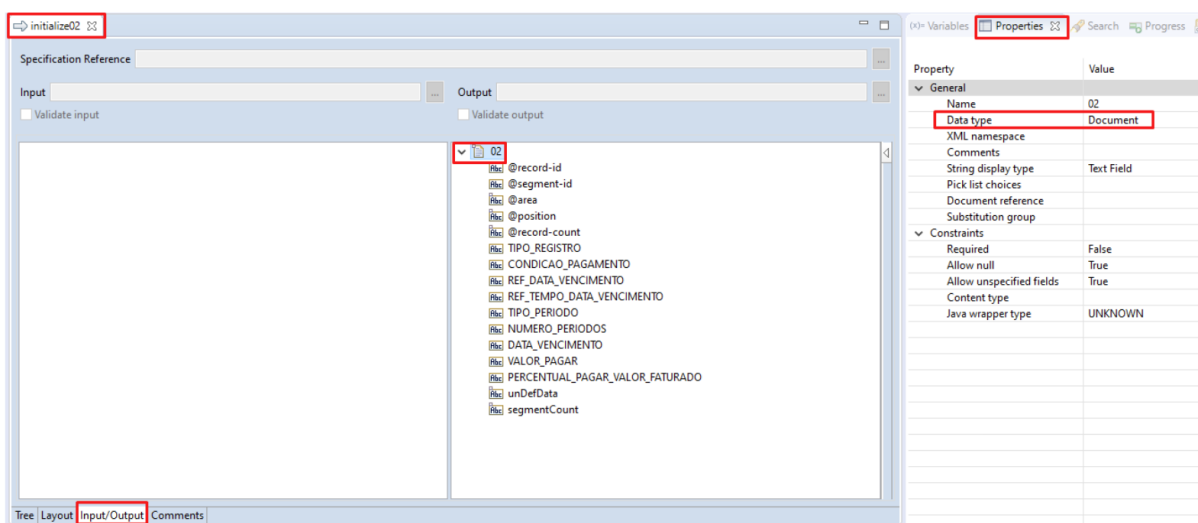


Figura 18 – Alterando ocorrência do registro para Document

Após criada a estrutura basta criar um MAP no Tree e inicializar os campos com vazio, para isto deve clicar duas vezes no campo e clicar em OK, porém para os campos obrigatórios @record-id e @segment-id devem ser inicializados com o Identificador do Registro, para isto deve clicar duas vezes no campo, inserir o valor desejado e clicar em OK.

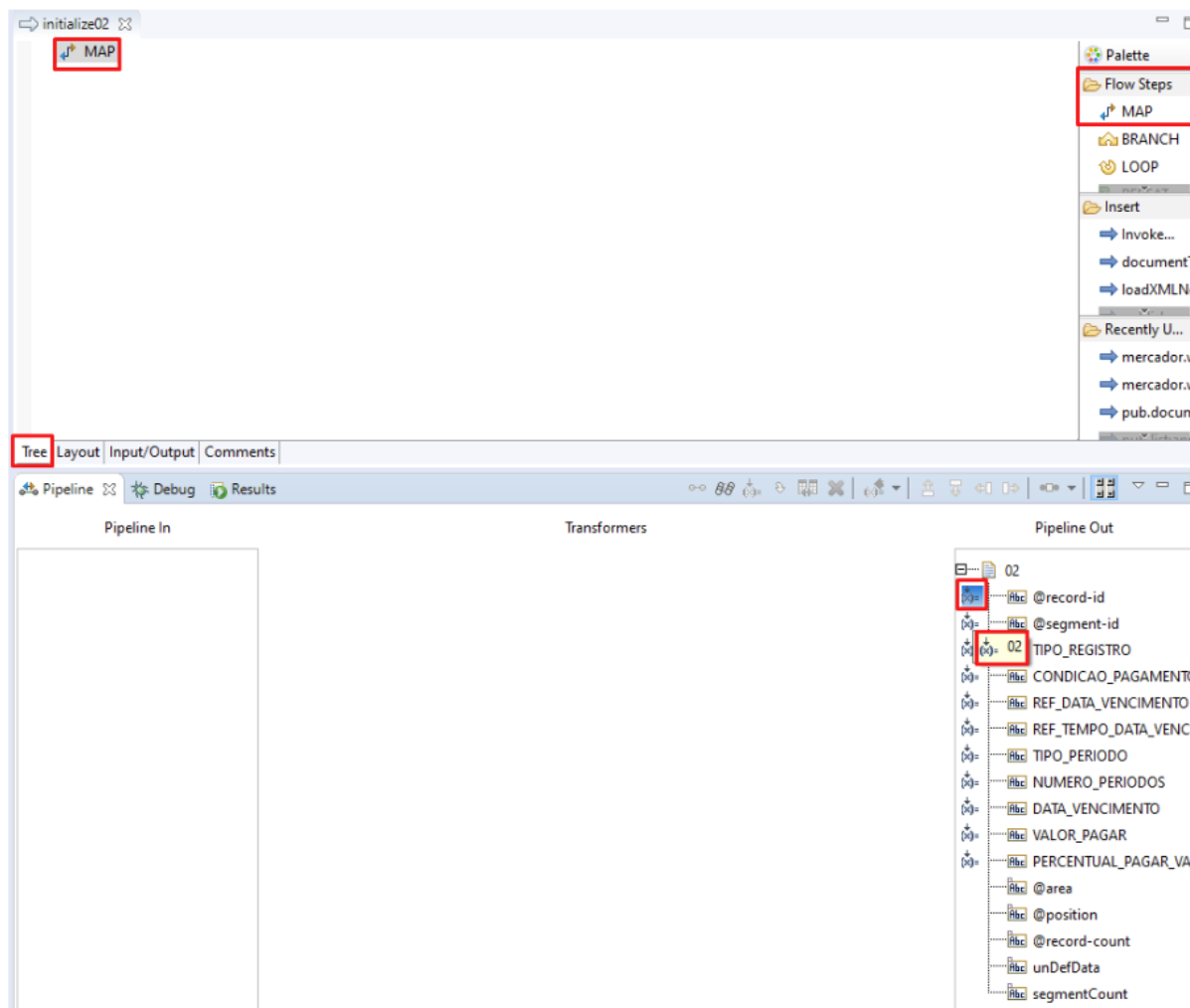


Figura 19 – Inicializando campos no Initialize array

Diferente do initialize de registros únicos, o initialize de array deve ser inicializado dentro do Loop referente a este registro no mapping. Conforme imagem abaixo:

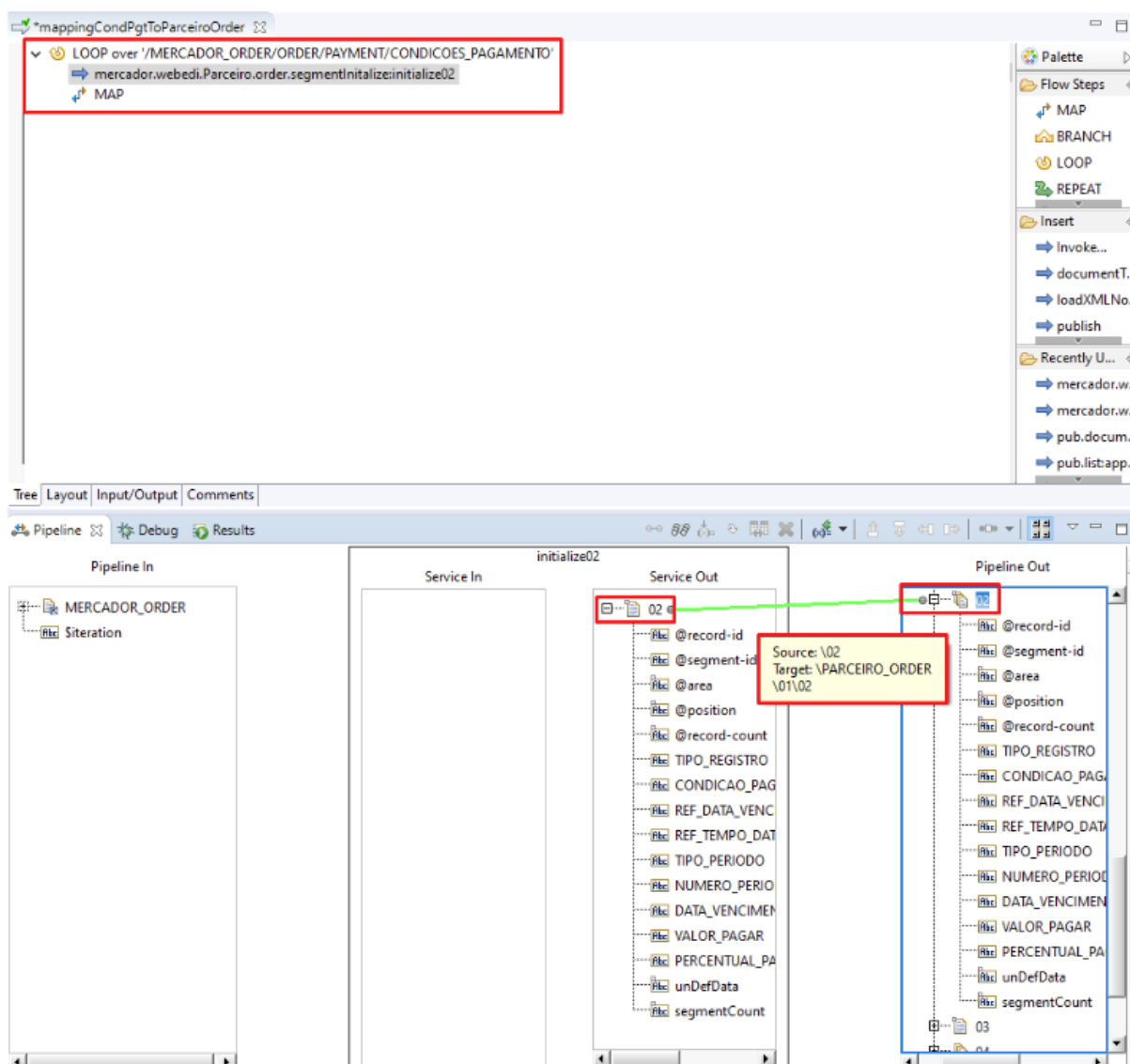


Figura 20 – Usando mapeando o initialize array para o registro dentro do LOOP

7 - Configurando os mapeamentos

Para vias de organização, os mapping únicos dos mapping de array ficam em dois serviços diferentes. Registros únicos são os registros que recebem uma única vez o valor no mapeamento nos seus campos. O registro único possui Ocorrência Máxima de 1 registro. Registros array são registros que possuem campos que se repetem. Pode haver um ou mais Mapeamentos de Array no mesmo layout. O registro array não possui Ocorrência Máxima.

7.1 - Realizando o mapeamento

Primeiramente, deve-se adicionar o template do mercador em INPUT e o layout

que está na pasta records em OUTPUT. A partir disso, os mapeamentos podem ser desenvolvidos.

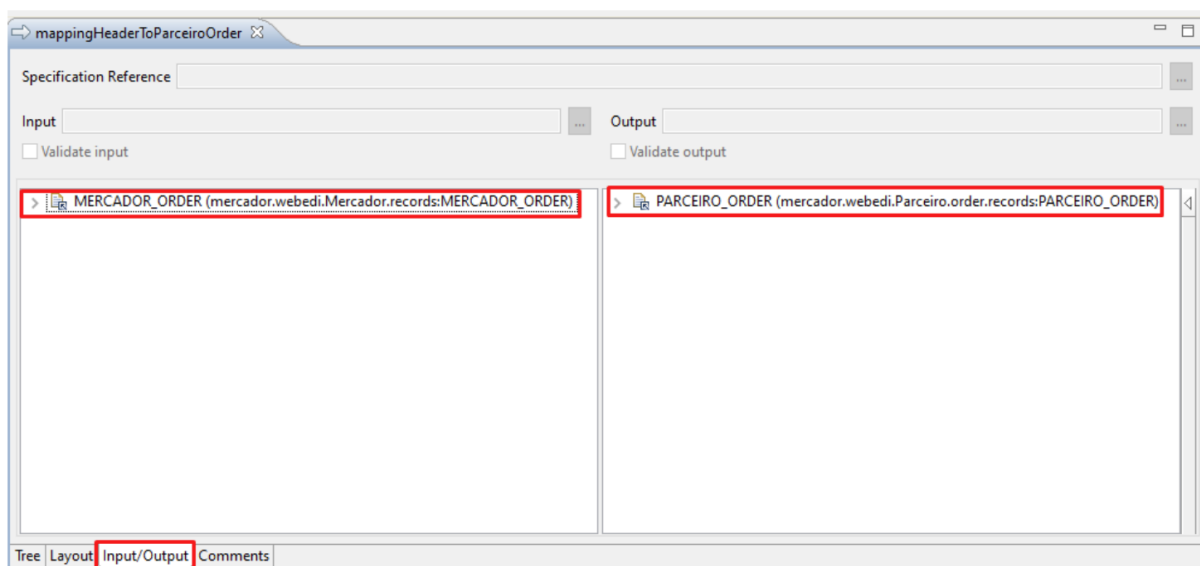


Figura 21 – Configurando o Input/Output dos mapeamentos

8 - Configurando o Adjust

Este serviço serve para ajustar o tamanho de cada campo na saída. Os campos devem conter exatamente o número de posições que está definido na plataforma de armazenamento de layouts de clientes, mesmo quando os campos não estiverem mapeados ou quando os valores não conterem o número total de posições. Isso ocorre porque o layout do cliente deseja receber o arquivo como combinado.

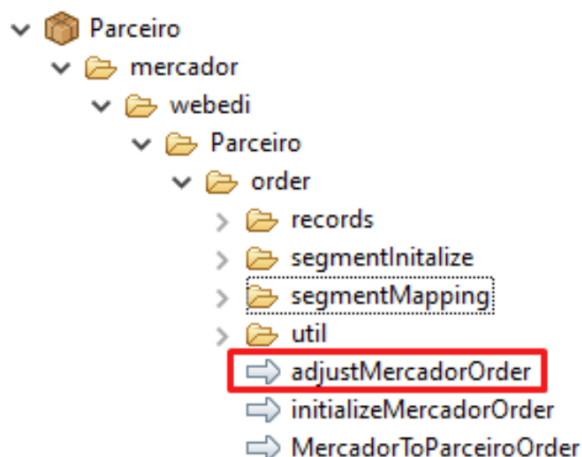


Figura 22 – Configurando o Adjust

Primeiramente, deve-se adicionar o layout que está na pasta records em INPUT

com a nomenclatura finalizada com `_IN` e o layout que está na pasta `records` em `OUTPUT` com a nomenclatura finalizada com `_OUT`. A partir disso, os próximos passos poderão ser realizados.

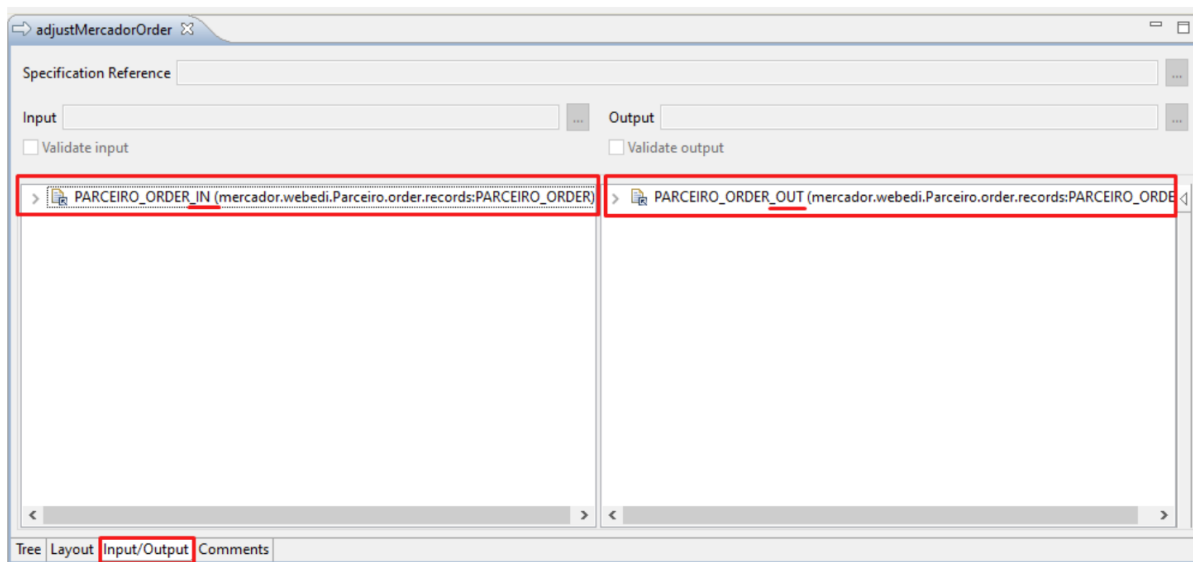


Figura 23 – Configurando o Input/Output do Adjust

Depois disso, basta mapear o campo do `template_IN` para o campo do `template_OUT` utilizando o `adjust` necessário, são eles `mercador.webedi.adjustAlfa:adjustAlfa` (para os campos de formato Alfanumérico e campos que contenham informação de data/hora) e `mercador.webedi.adjustNumeric:adjustNumeric` (para os campos de formato Numérico).

O `adjust` deverá ser escolhido de acordo com o tamanho e quantidade de decimais do campo. Por exemplo: um campo de formato Numérico com tamanho 15 e quantidade de decimais 2 no Mapping significa 15 dígitos sendo 2 deles decimais. Porém, o `adjust` a ser utilizado deve ser o `adjustNumeric13_2`, pois a interpretação no `adjust` é que são 13 dígitos e mais 2 decimais (equivalente aos 15 do gerenciador de layouts). Caso não exista um `AdjustNumeric` com a precisão desejada, usa-se o `mercador.webedi.Mercador:adjustNumber`. Para os registros em array, é necessário fazer os `LOOPS` necessários para tratar todos os campos.

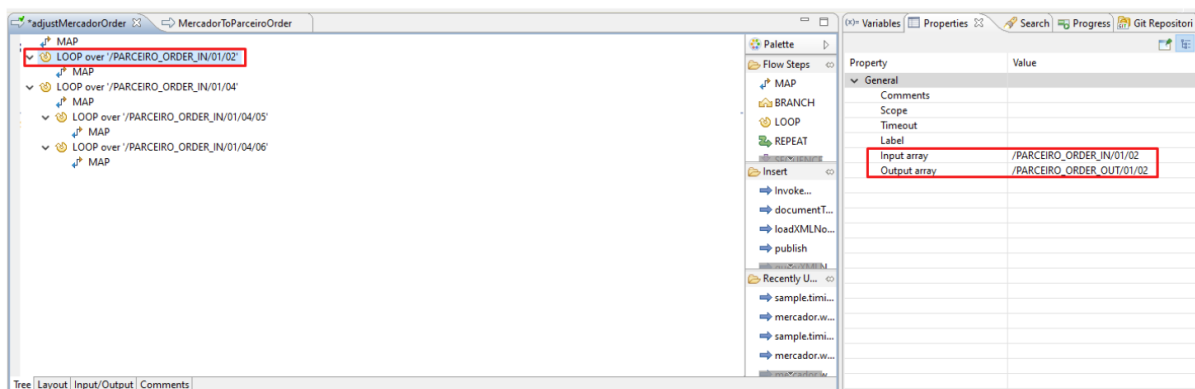


Figura 24 – Configurando LOOPS no Adjust

9 - Configurar o Serviço Principal

O serviço principal, depois de ter passado da primeira etapa de configuração, deve se parecer com o seguinte:

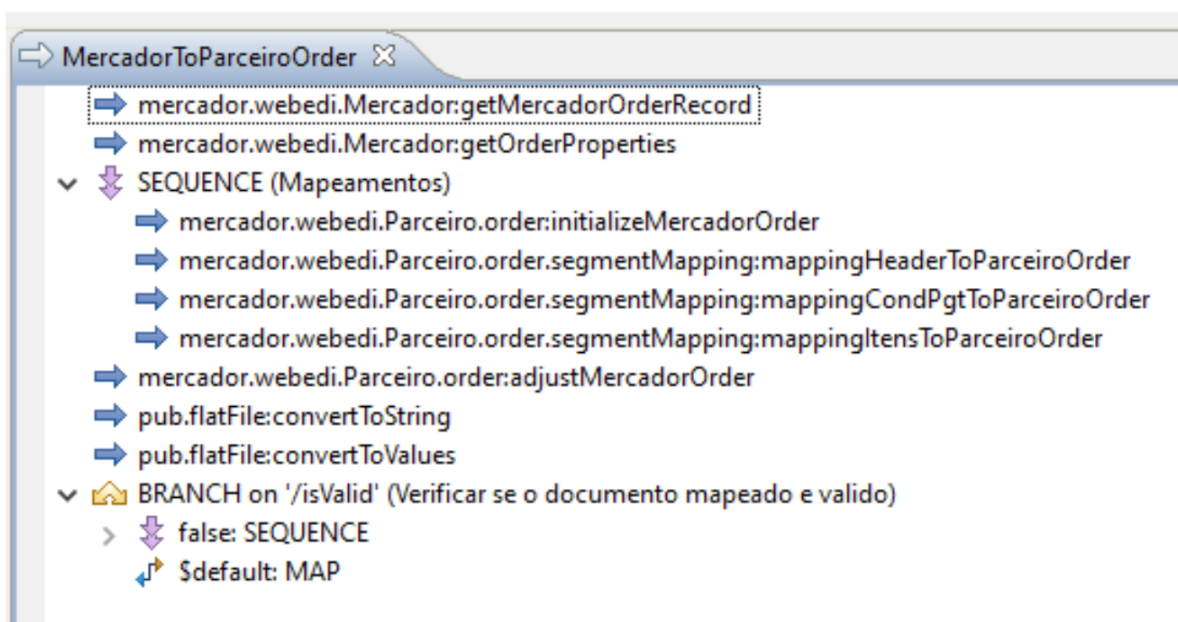


Figura 25 – Serviço principal após adição dos outros serviços desenvolvidos

Para finalizar o Serviço de Exportação, é necessário ajustar o serviço principal de acordo com o INPUT e OUTPUT adicionado nos serviços mapeados. No initialize o layout do cliente deve estar sendo mapeado do Service Out para o layout do cliente do Pipeline Out:

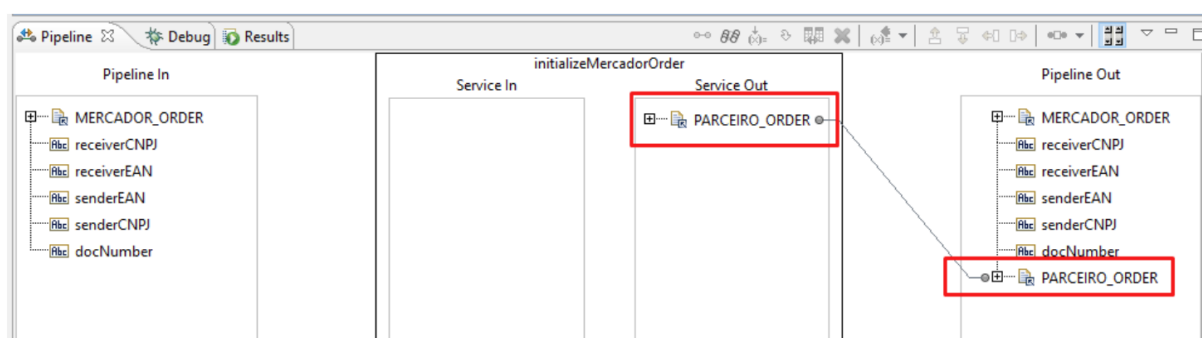


Figura 26 – Initialize para campos únicos

Os mapping devem estar sendo mapeados da seguinte forma: Layout Padrão Mercador do Pipeline In mapeado para o Layout Padrão Mercador do Service In; Layout do cliente do Service Out mapeado para o Layout do cliente do Pipeline Out.

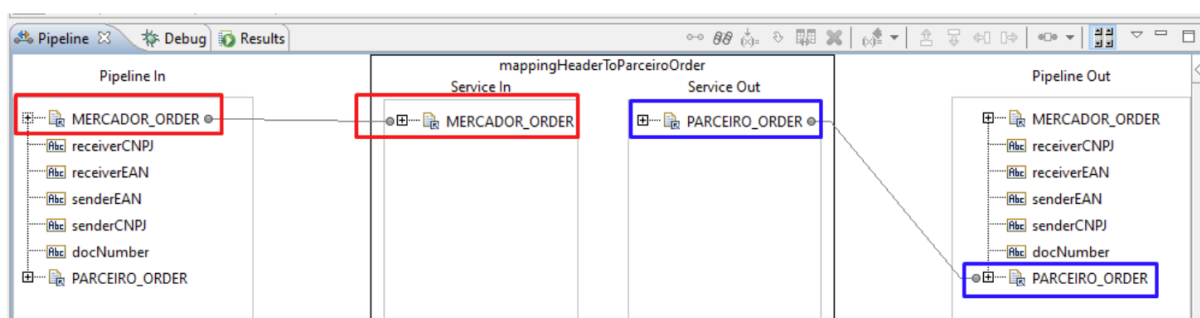


Figura 27 – Configurando pipeline do mapping único

No adjust precisa mapear o Layout do cliente do Pipeline In para o Layout do cliente IN do Service In e depois o Layout do cliente OUT do Service Out para o Layout do cliente do Pipeline Out.

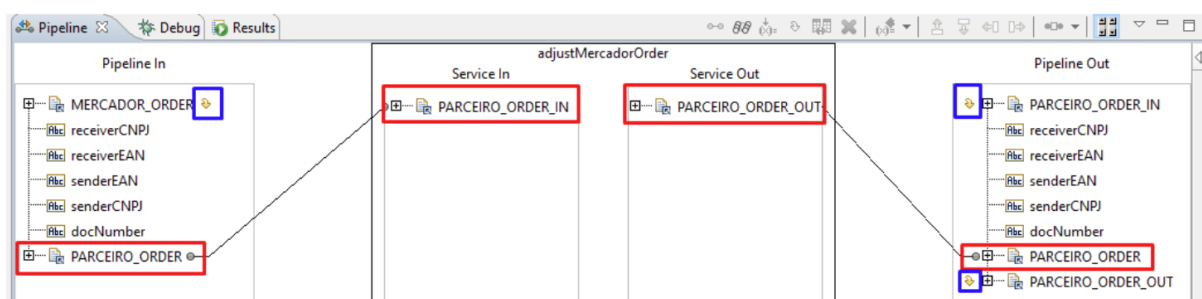


Figura 28 – Configurando pipeline do Adjust

No convertToString é necessário mapear o Layout do cliente do Pipeline In para o ffValues do Service In e fixar o Schema no ffSchema do Service In.

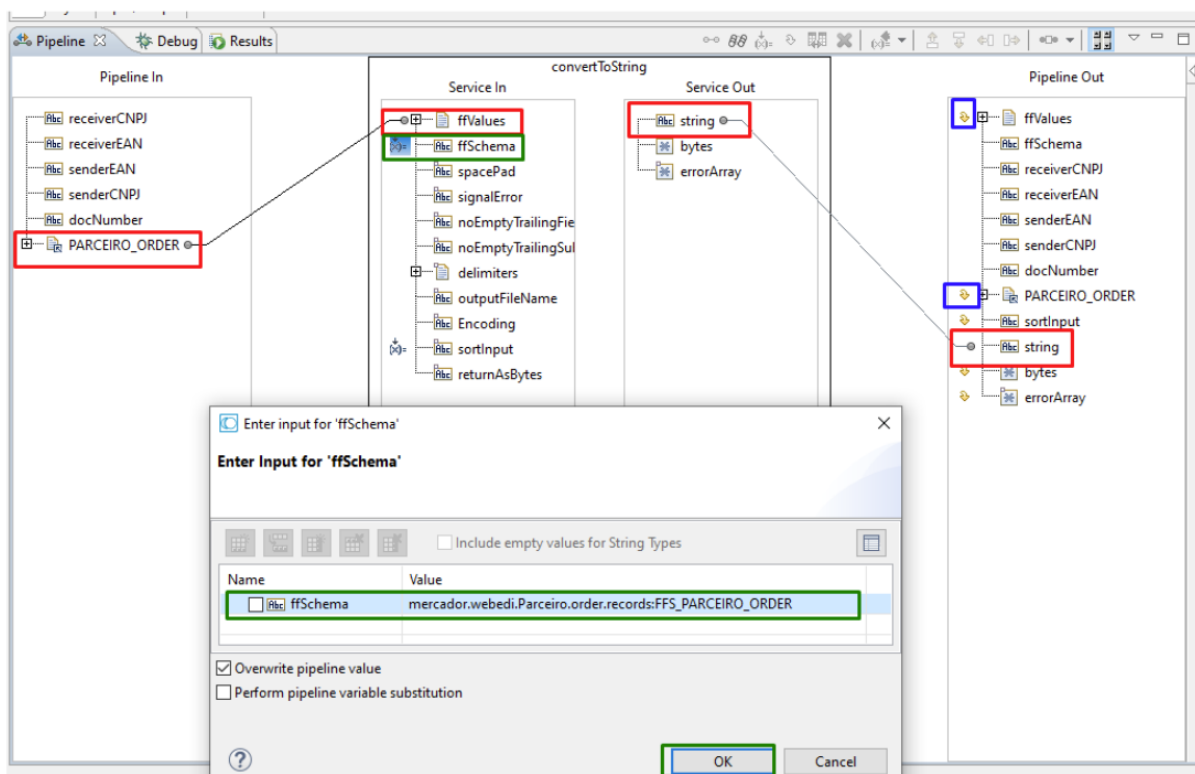


Figura 29 – Configurando serviço convertToString

No convertToValues é necessário mapear a string do Pipeline In para o ffData do Service In, e após dropar no Pipeline Out o ffData, ffSchema, ffValues e fflteator.

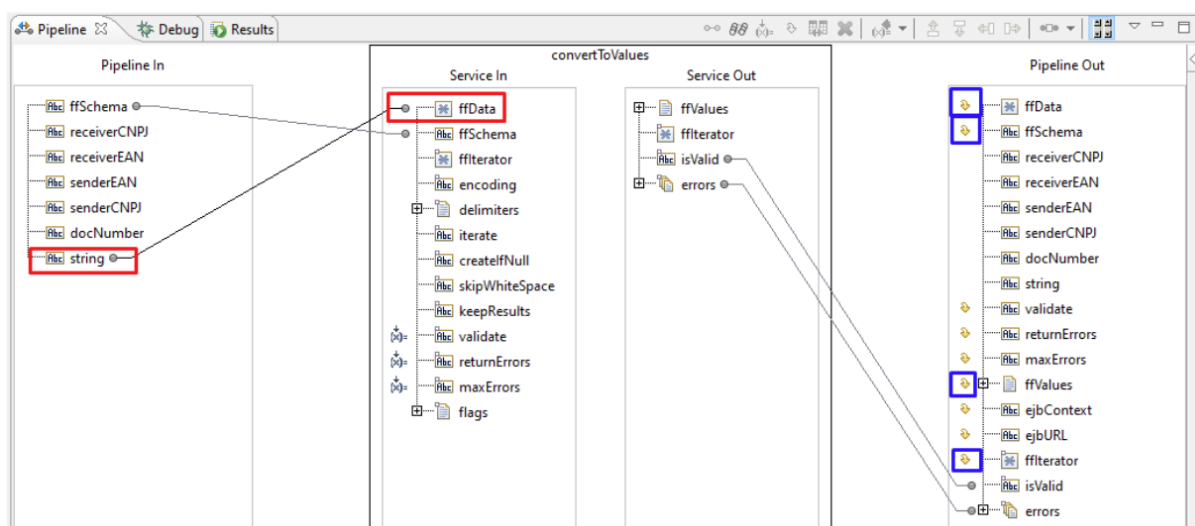


Figura 30 – Configurando serviço convertToValues

Depois deste desenvolvimento devemos testar se o serviço está rodando como o esperado, ou seja, efetuando a tradução do layout XML para o Flat File

3.3.3 Conectividade

Nesta atividade, conduzimos conexões fundamentais com clientes que buscavam enviar arquivos à Neogrid para posterior processamento, gerando relatórios, gráficos e alertas de inteligência de mercado. Essas análises eram cruciais tanto para o contratante, que se beneficiava das informações estratégicas, quanto para a comunicação entre as pontas envolvidas no processo.

Tomando como exemplo a situação previamente mencionada da Kabum, que realiza pedidos de produtos junto à Apple. Para que esse pedido se concretize, é essencial que haja uma comunicação eficaz não apenas entre a Kabum e a Neogrid, mas também entre a Neogrid e a Apple. Nesse fluxo, a Neogrid atua como intermediária, recebendo dados de um lado e enviando-os para o outro, desempenhando um papel crucial na eficiência da transação comercial.

A escolha da conectividade é uma decisão flexível e personalizada, à critério do cliente. A Neogrid oferece uma variedade de opções, tais como API Rest, SFTP, FTP, FTPS, AS2, NGClient (proprietário) e Webservice SOAP. Essa diversidade permite adaptabilidade às preferências e necessidades específicas de cada cliente.

Uma vez estabelecido o contrato, a demanda é encaminhada para o time de Ingestão de Dados. Nessa etapa, a equipe desenvolve a conectividade necessária utilizando a plataforma WebMethods, menos quando se trata de NGClient, pois ele é uma aplicação separada.

Irei descrever abaixo como é feito o desenvolvimento de um serviço SFTP de envio de arquivos para o cliente, ou seja, iremos enviar um arquivo para um servidor SFTP.

3.3.4 Passo a Passo

1 - Criando pastas

- Criar um pacote com o nome do cliente + Conectividade (Exemplo: ParceiroConectividade)
- Criar uma pasta mercador
- Criar uma pasta webedi
- Criar uma pasta com o nome do cliente (Exemplo: Parceiro)
- Criar uma pasta com o nome da conectividade (Exemplo: sftp)

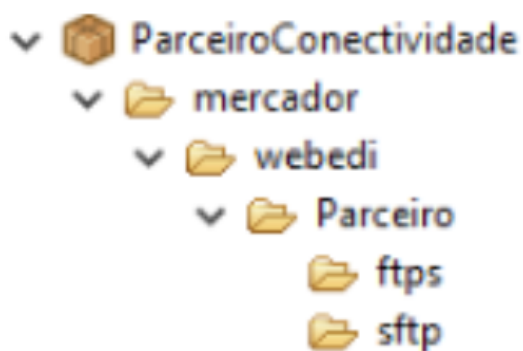


Figura 31 – Pacote e pastas

2 - Templates

No pacote de Templates, basta copiar o conteúdo da pasta da conectividade solicitada para o pacote criado e realizar as alterações necessárias, substituindo Template do nome do serviço para o nome do cliente (Exemplo: Parceiro).

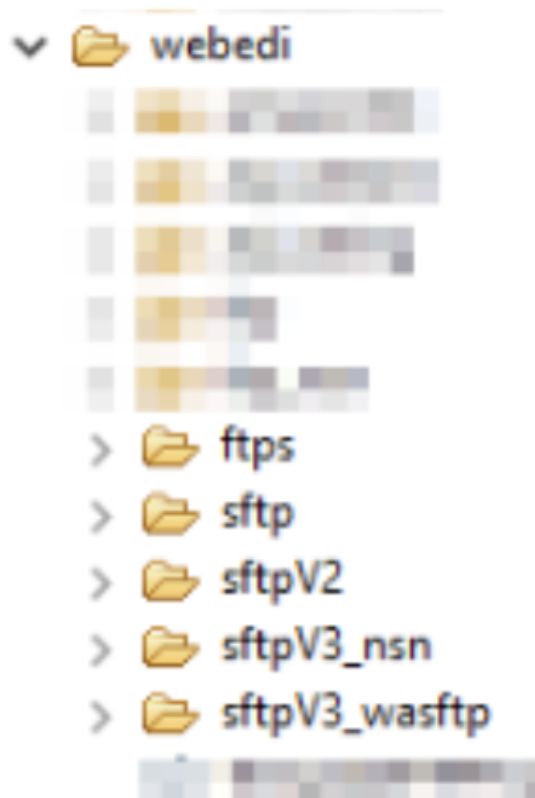


Figura 32 – Templates de SFTP

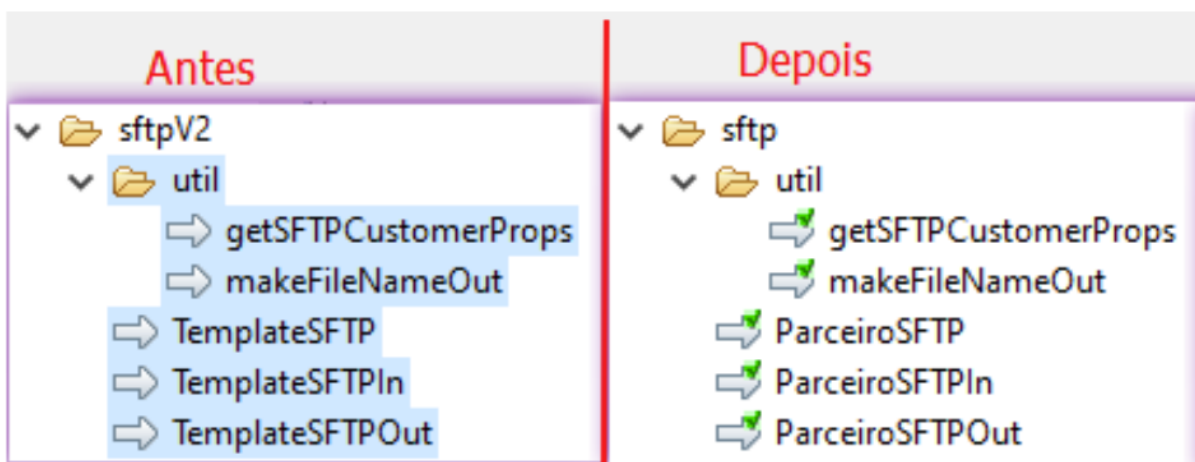


Figura 33 – Adaptando nomenclatura dos serviços

3 - Criação do Server Alias e User Alias no Portal

Acessar o painel WmRoot do ambiente em que está sendo desenvolvido o SFTP, ir em Settings > SFTP

A captura de tela mostra o painel de administração do WmRoot. No topo, há o cabeçalho 'WEBMETHODS Integration Server' e o caminho de navegação 'Settings > SFTP'. Abaixo, há links para 'Create Server Alias' e 'User Alias Settings'. O principal elemento é a 'SFTP Server List', uma tabela com as seguintes colunas: Alias, Host, Port, Proxy Alias, Host Key Fingerprint e Delete. A tabela contém 20 linhas de dados, cada uma representando um servidor SFTP configurado.

Alias	Host	Port	Proxy Alias	Host Key Fingerprint	Delete
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗
...	✗

Figura 34 – Configurando alias do servidor e usuário

4 - Server Alias

Acessar Create Server Alias. Preencher o Alias, Host Name or IP Address e Port Number, após clicar em Get Host Key que irá gerar uma chave, agora é só salvar em Save Changes.

Settings > SFTP > Create Server Alias

[Return to SFTP](#)

SFTP Server Alias Properties

Alias	<input type="text" value="Parceiro_SFTP"/>
Host Name or IP Address	<input type="text" value="host.com.br"/>
Port Number	<input type="text" value="22"/>
Preferred Key Exchange Algorithms	<div><ul style="list-style-type: none">ecdh-sha2-nistp256ecdh-sha2-nistp384ecdh-sha2-nistp521diffie-hellman-group14-sha1diffie-hellman-group-exchange-sha256diffie-hellman-group-exchange-sha1diffie-hellman-group1-sha1</div> <div><input type="button" value="Up"/> <input type="button" value="Down"/></div>
Proxy Alias	<input type="text"/>
Host Key Location	<input type="text"/> <input type="button" value="Get Host Key"/>

Figura 35 – Criando alias do servidor

5 - User Alias

Acessar o User Alias Settings e depois Create User Alias. Preencher Alias, User Name, selecionar o Authentication Type (Password ou Public Key), se selecionar Password preencher Password e Re-type Password; se selecionar Public Key preencher o Private Key Location (arquivo deve estar em um diretório), PassPhrase e Re-type PassPhrase. Selecionar SFTP Server Alias que criou na etapa anterior e salvar em Save Changes.

Settings > SFTP > User Alias Settings > Create User Alias

- Return to SFTP
- Return to User Alias Settings

SFTP User Alias Properties

Alias:

User Name:

Authentication Type: Password Public Key

Password:

Re-type Password:

SFTP Server Alias:

Authentication Type: Password Public Key

Private Key Location:

PassPhrase:

Re-type PassPhrase:

SFTP User Alias Advanced Settings (Optional)

Maximum Retries:

Connection Timeout: seconds

Session Timeout: minutes

Strict Host Key Checking: Yes No

Compression:

Compression Level:

Figura 36 – Criando alias do usuário

Para validar se a autenticação foi preenchida corretamente, clique no botão de play da coluna Test.

SFTP User List					
SFTP User Alias	SFTP Server Alias	User	Authentication Type	Test	Delete
<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input checked="" type="button" value="▶"/>	<input type="button" value="✖"/>

Figura 37 – Testando conexão SFTP

6 - Configuração no webMethods

Depois abrir o serviço getSFTPCustomerProps que consta dentro da pasta util e preencher as variáveis abaixo:

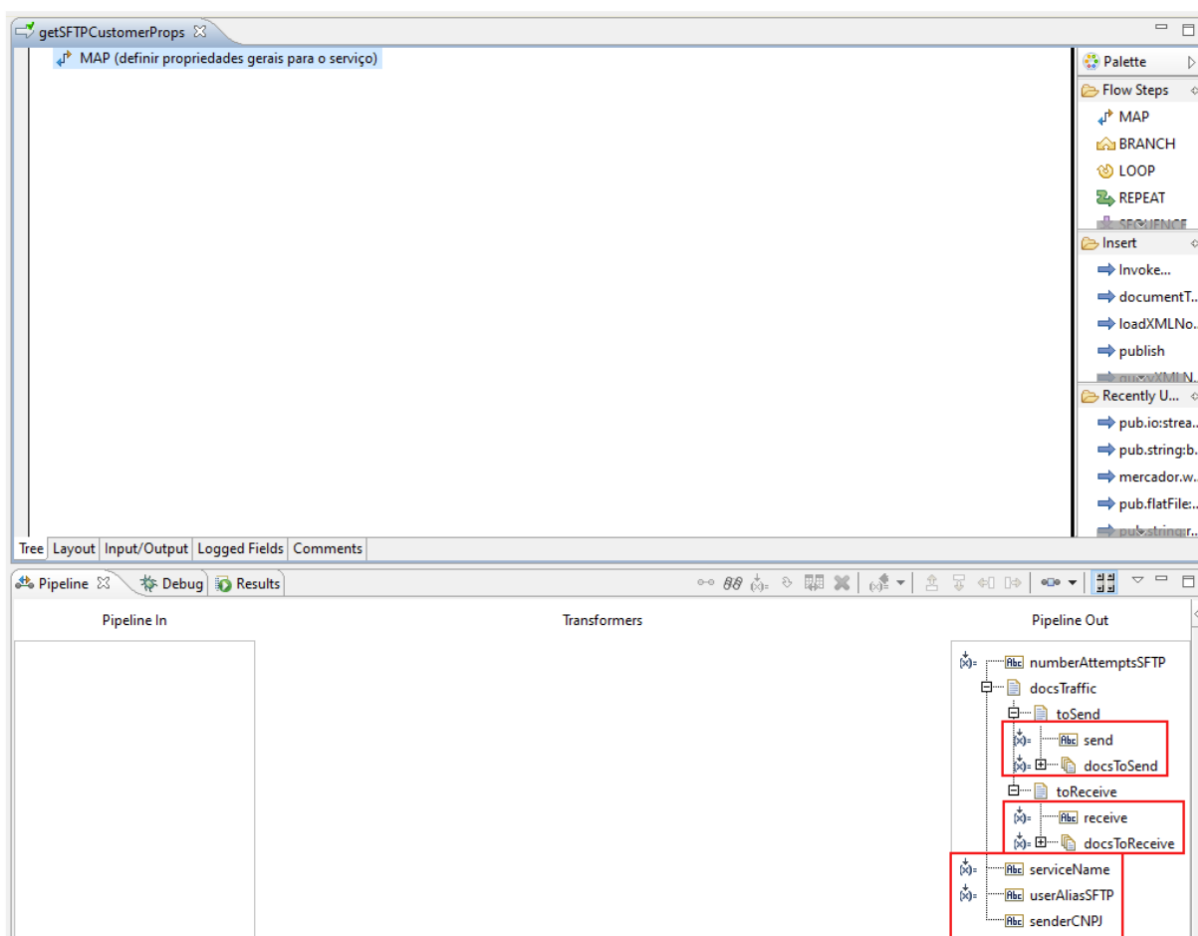


Figura 38 – Preenchendo variáveis do serviço getSFTPCustomerProps

3.4 Projetos

3.4.1 EDP

Este foi um projeto que aconteceu entre o terceiro e quarto trimestre de 2022. Ele consistia em efetuar a migração do projeto que estava desenvolvido em outro SaaS contratado pela Neogrid para o webMethods. Nesta migração, o time de Ingestão de Dados precisou desenvolver todo o fluxo de conectividade e transformação do dado desde seu início. Foram desenvolvidos 11 integrações para o tráfego de documentos fiscais.

Na área de conectividade, para ter acesso aos dados que precisavam ser tratados dentro da Neogrid era necessário para 6 documentos fiscais que a efetuássemos requisições na API REST do cliente, assim a partir do retorno, montar os arquivos e inserí-los numa aplicação responsável por orquestrar a messengeria até um dos produtos Neogrid. Para os outros 5 o cliente chamava um endpoint de uma API REST Neogrid e enviava informações que eram utilizadas para montar estes arquivos e colocá-los na aplicação. Depois da conectividade capturar os arquivos e inserir eles na aplicação,

era a vez da transformação dos arquivos. Os arquivos chegavam no formato JSON e a solução Neogrid contratada pelo cliente necessitava receber o conteúdo em CSV. Minha atuação foi de extrema importância neste projeto, fiquei responsável por desenvolver e projetar os onze tradutores.

No início do projeto, meu desejo era criar somente um tradutor que efetuasse a transformação de todos documentos fiscais não importando qual passasse pelo tradutor, porém por questões de performance isto não foi possível, por fim foi necessário desenvolver 11 tradutores individuais, o que tornou o desempenho melhor e facilitou o processo de manutenção, caso necessário.

3.4.2 Berlanda

O projeto Berlanda começou no quarto trimestre de 2023, ele se tratou de uma venda do produto DRP que trouxe um faturamento bastante significativo para a Neogrid. O time de Ingestão de dados ficou encarregado de desenvolver o consumo do Webservice do cliente, transformar o dado num formato compatível para o produto e então enviá-lo a ele. Foi necessário o desenvolvimento de 13 fluxos de integração para cada tipo de documento que era preciso ser integrado no produto.

Na área de conectividade tivemos um desafio quanto ao Webservice do cliente, pois ele era de desenvolvimento do cliente, portanto não seguia as melhores práticas de mercado, o que tornou necessária a criação de alguns contornos para conseguirmos extrair os documentos da infraestrutura do cliente.

No âmbito das transformações, o cliente gerava um arquivo XML pelo seu Webservice e o produto Neogrid era capaz de receber o formato CSV em layouts com campos pré-definidos, então foi necessário transformar o arquivo e preencher cada campo conforme o layout definido pelo produto. Participei no desenvolvimento de uma padronização de desenvolvimento para os 13 serviços de tradução, ou seja, eu defini como seria a estrutura dos serviços do forma geral, além de ter desenvolvido alguns deles.

4 RELAÇÃO DAS ATIVIDADES COM OS COMPONENTES CURRICULARES E APRENDIZADO NO CURSO

A disciplina de Engenharia de Software (3º semestre) foi essencial para compreender e aplicar metodologias ágeis no ambiente corporativo. Na empresa, adotamos práticas do Kanban, como o uso de um quadro com cards que representam cada atividade e colunas verticais que indicam as etapas de progresso desses cards. Isso permitiu uma organização clara e visual das tarefas, facilitando o acompanhamento de status e o gerenciamento do fluxo de trabalho. Além disso, implementamos uma prática ágil conhecida como "daily meeting"(reunião diária), um momento no qual a equipe se reúne para compartilhar obstáculos enfrentados nas atividades. Essa prática ajudou a criar um ambiente colaborativo, onde problemas eram discutidos em equipe para encontrar soluções.

A disciplina Desenvolvimento II (5º semestre) também foi relevante, especialmente na experiência de trabalho em grupo com colegas que possuem diferentes habilidades e modos de trabalho. Esse contato direto com as dinâmicas de equipe, em um ambiente controlado, me preparou para lidar com situações reais no ambiente corporativo, onde o sucesso de um projeto depende da capacidade de colaborar, adaptar-se e respeitar as peculiaridades de cada membro do time.

A lógica de programação, que está embutida nas disciplinas de Programação I, II e III (1º, 2º e 3º semestre) e Estrutura de Dados (2º semestre), aprendida na disciplina inicial do curso, é essencial para estruturar algoritmos. No meu ambiente profissional, a habilidade de criar algoritmos performáticos e eficientes tornou-se crucial na resolução de problemas cotidianos em serviços de transformação.

Os conceitos básicos de Banco de dados adquiridos durante a disciplina Banco de Dados I (2º semestre) e Banco de Dados II (3º semestre) foram aplicados com frequência no desenvolvimento de serviços de transformação. Consultas SQL são essenciais para os serviços de transformação, pois podemos buscar dados em

diferentes bases para enriquecer um arquivo do cliente. Tendo este conhecimento posso garantir que estas consultas não impactam na performance daquele serviço, pois se isso ocorrer podemos ter lentidão na operação da Neogrid.

Redes de Computadores (4º semestre) proporcionou uma compreensão geral da camada de aplicação do Modelo OSI. Essa base teórica é aplicada na configuração e manutenção de serviços de conectividade, assegurando a comunicação eficiente entre os sistemas da Neogrid e do cliente.

Os conceitos de Arquitetura Orientada a Serviços (SOA) aprendidos na disciplina de Gestão de Processos e Serviços foram importantes para compreender a estrutura e o funcionamento do WebMethods. A arquitetura SOA, baseada na ideia de modularidade e interoperabilidade entre sistemas, ajudou a entender como o WebMethods organiza seus serviços de forma independente e reutilizável, proporcionando flexibilidade e escalabilidade na integração entre a Neogrid e sistemas empresariais do clientes. No WebMethods, a abordagem SOA é evidente na maneira como ele permite o desenvolvimento de serviços modulares que podem ser invocados independentemente, seja para transformação de dados ou conectividade. Isso está alinhado aos princípios da SOA, como desacoplamento e reusabilidade. Por exemplo, o uso de serviços Web (SOAP e REST) no WebMethods reflete o suporte à comunicação padronizada e à integração de sistemas heterogêneos, uma característica central da SOA.

5 CONCLUSÃO

Ao concluir este relatório sobre minha atuação como desenvolvedor júnior na Neogrid, percebe-se a importância de uma formação sólida e da aplicação prática dos conceitos adquiridos no curso de Análise e Desenvolvimento de Sistemas. A experiência proporcionou uma visão aprofundada do setor de tecnologia e especificamente gestão de cadeias de suprimentos, onde a integração de dados e a automação de processos são elementos-chave. O uso da plataforma WebMethods, aliado aos conhecimentos em programação, banco de dados e redes de computadores, permitiu uma atuação eficaz nas áreas de transformação e conectividade de dados, garantindo a de informações e a comunicação entre sistemas diversos.

As metodologias ágeis, aprendidas nas disciplinas do curso, foram utilizadas para a organização das atividades e para a colaboração com a equipe. O contato com elementos dessas metodologias na prática demonstrou a importância da plasticidade e da comunicação constante em ambientes corporativos. Dessa forma, a experiência profissional não apenas solidificou o aprendizado teórico, mas também evidenciou como o curso prepara para os desafios técnicos e interpessoais do mercado de trabalho. Essa experiência contribuiu fortemente para meu desenvolvimento profissional e reforçou a relevância dos componentes curriculares no preparo para uma carreira em tecnologia da informação.

Pelo o que o curso de Análise de Desenvolvimento de Sistemas nos proporciona nos 6 semestres, na minha opinião, ele não deixou nada a desejar para a posição que ocupei como desenvolvedor Júnior nestes 2 anos descritos no relatório. Os conceitos gerais de áreas como arquitetura de computadores, sistemas em nuvem, processamento de dados, etc. foram transmitidos nas disciplinas do curso e ao trabalhar na Neogrid estes conceitos precisaram ser aprofundados e ligeiramente moldados para o cenário da Neogrid.

REFERÊNCIAS

Andrew Warner. **How Caren Merrick started WebMethods in her basement and helped build it to a public company that sold for \$546 million.**

2016. [Acessado em: 25 de novembro de 2025]. Disponível em: <<https://mixergy.com/interviews/webmethods-with-caren-merrick/>>.

Casares Pinto, O., Monteiro Escott, C., Weber, T. *et al.* **Projeto Pedagógico do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas.** Porto Alegre (RS), 2017. Disponível em: <https://restinga.ifrs.edu.br/site/midias/arquivos/20171122104136422projeto_pedagoogico_curso_ads_resolucao_101_consulpdf>.

David Rocks. **Phillip Merrick, CEO, webMethods.** 2000. [Acessado em: 25 de novembro de 2025]. Disponível em: <https://web.archive.org/web/20160329213707/http://www.businessweek.com/2000/00_20/b3681088.htm#expand>.

Neogrid. **Neogrid - Soluções de ponta a ponta para a gestão da sua empresa.** 2024. [Acessado em: 27 de outubro de 2024]. Disponível em: <<https://www.neogrid.com>>.