

---

## Mirth Connect: Introdução



Com o advento das tecnologias digitais e dos dispositivos conectados, a integração ainda continua sendo um desafio na atenção à saúde. Para oferecer um melhor atendimento ao paciente, os dados eletrônicos sobre sua saúde, precisam fluir entre fronteiras e diferentes sistemas.

No entanto, ainda existem fatores como a relutância das instituições de saúde sobre o compartilhamento dos dados, a baixíssima adesão a padronização dos dados, os silos de dados proprietários gerados pelos EHRs e os requisitos de privacidade e segurança, fatores estes, que tornam impossível para que diferentes sistemas possam intercambiar dados eletrônicos em saúde, por si e entre si.

Embora tenham sido projetadas ontologias, padrões para os dados e padrões para as mensagens, estes não foram tão bem sucedidos em relação ao aumento da eficiência e efetividade da integração de aplicativos, como originalmente pretendido.

À medida em que os padrões evoluíram, a complexidade da integração também aumentou. Manter novos padrões e os mecanismos de interfaces tornou-se um enorme fardo e um alto custo para as organizações de saúde. **Assim nasce o "Mirth"**

## O que é o Mirth?



Imagem: Vigyanix

O Mirth é uma plataforma que conecta diversos e diferentes sistemas eletrônicos em saúde para assim criar um ambiente interoperável mais rápido, fácil, seguro e econômico para a troca de informações em saúde.

O Mirth foi projetado como forma de tornar a integração entre os sistemas eletrônicos de saúde mais fácil e rápido, pois aceita os principais padrões de mensagens como HL7, FHIR, CDA, EDI, X12 e outros como PDF, CSV e etc.

Para isso, o Mirth Connect tem sua arquitetura baseada em canais os quais conectam os diferentes sistemas eletrônicos em saúde, tanto os internos como os externos, onde são aplicados os filtros, transformações e encaminhamento das mensagens através de regras definidas pela organização.

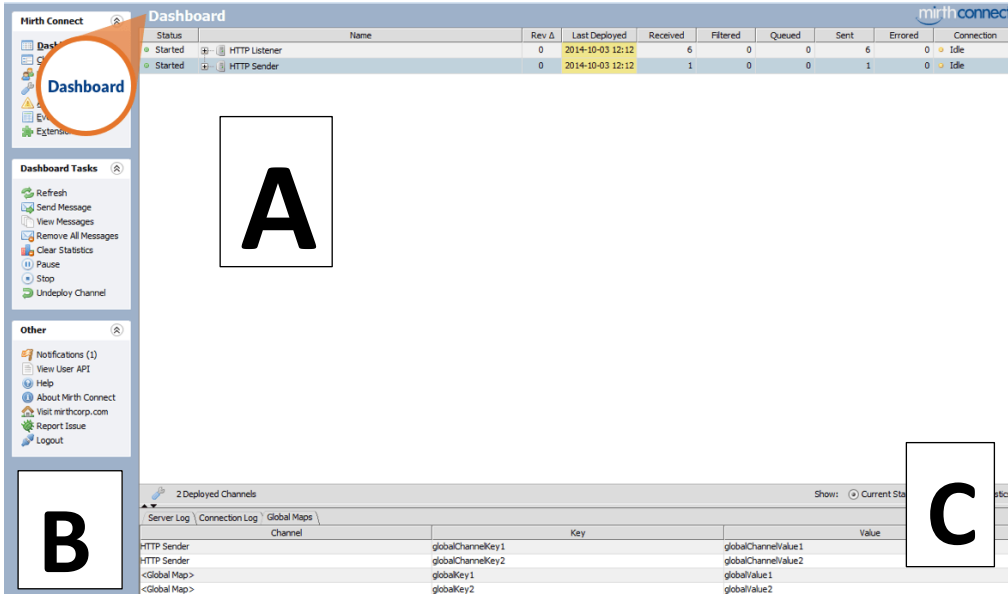
Vamos conhecer a arquitetura do Mirth e seus componentes:

### Arquitetura do Mirth

Existem quatro principais componentes na arquitetura do Mirth:

- **Servidor Mirth Connect:** interface de gerenciamento e componente do mecanismo de integração: realiza filtros, transformações e transmissão de mensagens
- **Administrador do Mirth Connect:** GUI que se conecta ao servidor do Mirth Connect. Permite configurar interfaces, monitorar a atividade das interfaces e realizar consultas no repositório das mensagens.
- **Gerenciador do Servidor de Conexão Mirth:** GUI que gerencia o Serviço Mirth Connect, exibe arquivos de log e outras configurações do Servidor Mirth Connect.
- **Interface de linha de comando do Mirth Connect:** uma ferramenta de linha de comando que permite que o servidor Mirth Connect seja implementado, importar e exportar canais e ainda executar outras tarefas administrativas.

## Interfaces do Mirth



The screenshot displays the Mirth Connect Dashboard. At the top, there is a table with the following data:

Status	Name	Rev Δ	Last Deployed	Received	Filtered	Queued	Sent	Errored	Connection
Started	HTTP Listener	0	2014-10-03 12:12	6	0	0	6	0	Idle
Started	HTTP Sender	0	2014-10-03 12:12	1	0	0	1	0	Idle

Below the table, there are three callout boxes labeled A, B, and C. Box A points to the 'Dashboard' link in the left sidebar. Box B points to the 'Dashboard Tasks' section in the left sidebar. Box C points to the '2 Deployed Channels' section at the bottom of the dashboard.

The '2 Deployed Channels' section contains a table with the following data:

Channel	Key	Value
HTTP Sender	globalChannelKey1	globalChannelValue1
HTTP Sender	globalChannelKey2	globalChannelValue2
<Global Map>	globalKey1	globalValue1
<Global Map>	globalKey2	globalValue2

Ao realizarmos o login no Mirth, somos levados ao painel de administração do Mirth Connect (Dashboard), que possui três seções:

#### A. Painel central

O painel exibe os "Canais":

- Último Deploy
- Estado dos canais
- Nome dos canais
- Canais implementados (ativos)
- Quantidade de mensagens recebidas (Filtradas, Na fila, Enviadas, Erros)

#### B. Menu de navegação à esquerda

#### C. Painel de logs na parte inferior

---

## Canais

A maior parte do trabalho no Mirth se faz através dos "Canais". Todas as mensagens são criadas e enviadas através dos Canais. O formato dos dados das mensagens também é definido aqui. O Canal está no menu de navegação do lado esquerdo (B).

Ao clicar no nome de um canal, expandirá o "Menu Tarefas do Canal" o qual incluem opções para edição, clonagem, exclusão e desativação de um Canal. Quando o usuário cria ou edita um canal existente, a página Editar Canal é exibida.

## Editar Canal

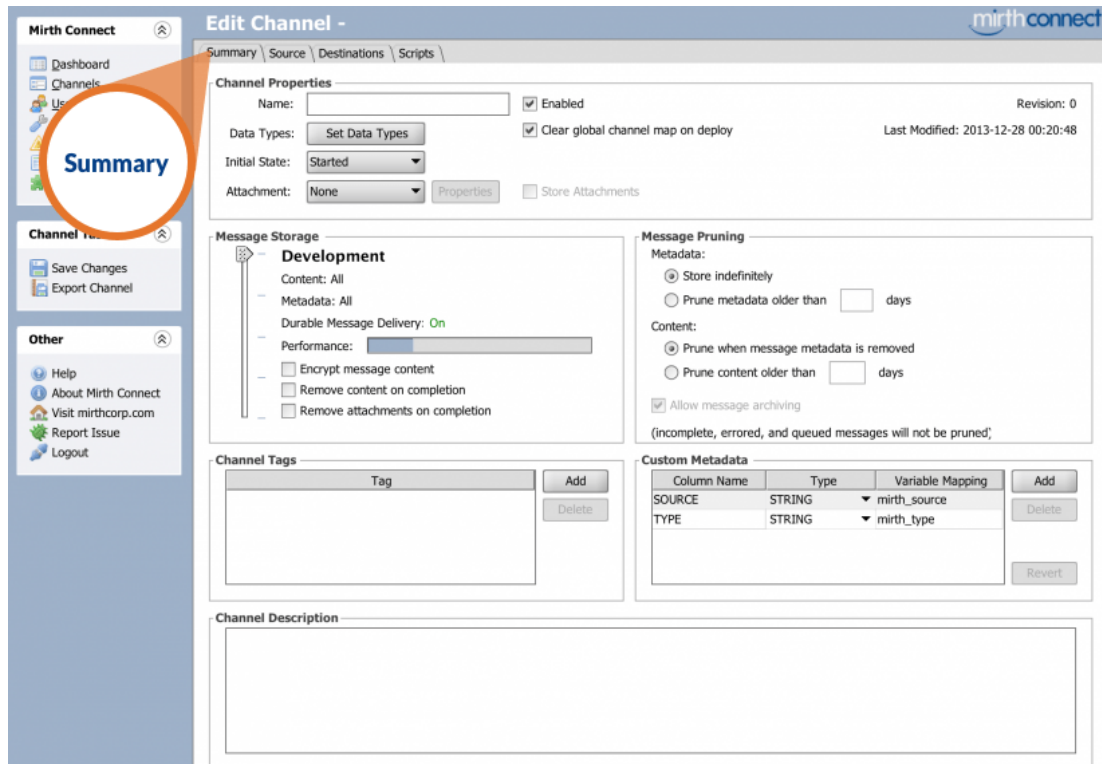
Possui quatro guias horizontais:

- A) Summary
- B) Source
- C) Destinations
- D) Scripts

Uma mensagem tem uma fonte e um ou mais destinos.

### Summary

Reúne informações gerais sobre o Canal como seu nome, descrição, data-types, etc., em quatro sessões.



### a. Channel Properties

Define as propriedades básicas para o Canal como seu Nome, ID, Tipos de dados (Permite ao usuário definir o tipo de dados recebidos e escolher entre os tipos de dados HL7, ASC (X12), XML, Texto Delimitado etc.).

### b. Message storage & pruning

Define como uma mensagem deve ser armazenada ou quando ela deve ser removida. Para armazenar as mensagens, o Mirth permite que você escolha entre 5 níveis diferentes:

- Desativado (sem armazenamento de mensagens)
- Metadados (armazena somente metadados, se houver)
- Raw (armazena mensagem em seu formato original (raw & meta data))
- Produção (armazena mensagens em formato raw, codificadas, enviadas, respostas e mapeamentos)
- Desenvolvimento (armazena tudo)

Os usuários podem selecionar o nível de armazenamento pelo scroller disponível no armazenamento de mensagens e também podem decidir se as mensagens recebidas pelo canal serão salvas ou não e, se salvas, devem ser criptografadas para que os dados privados do paciente sejam preservados e ainda, por quanto tempo as mensagens devem ser armazenadas no banco de dados antes de serem purgadas (removidas).

### c. Channel tags & metadata

Permite que adicionemos uma tag para o Canal, definir os metadados que desejamos salvar no Canal.

### d. Channel description

Adiciona uma descrição para o Canal.

---

## Source

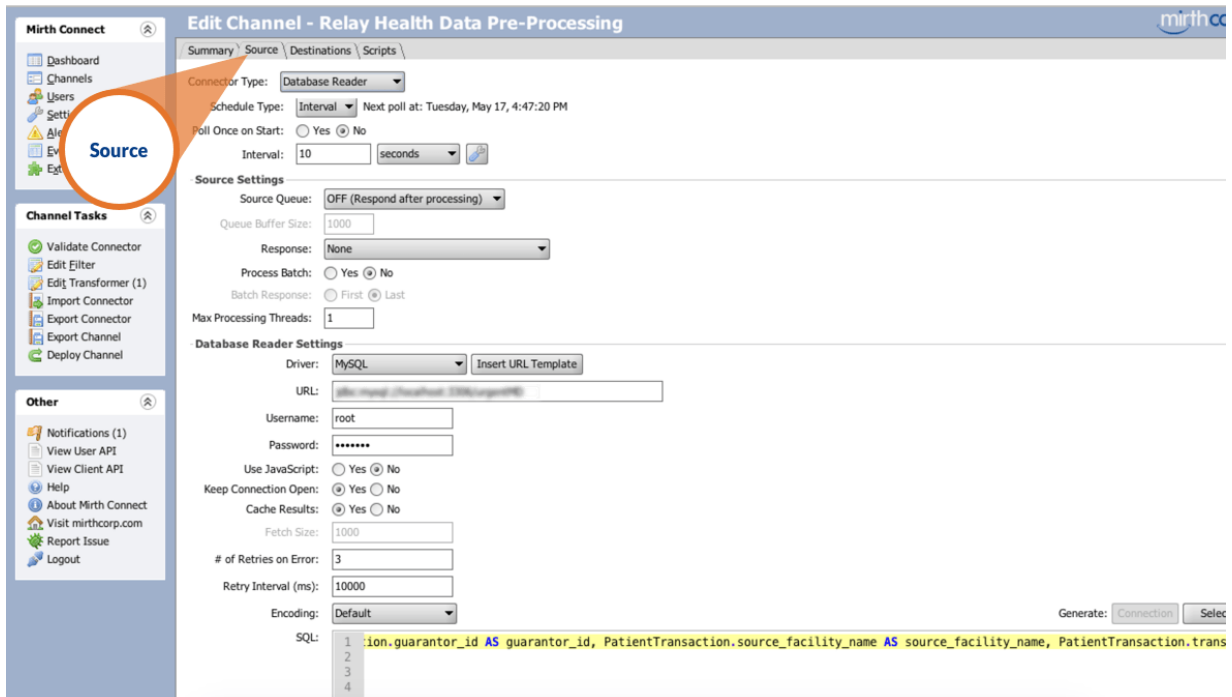
A guia "Source" permite definir como o Canal receberá os dados.

Na parte superior o usuário pode selecionar o tipo de conector com o qual vai trabalhar. Listamos os principais conectores sem ordem específica:

- HTTP Listener
- LLP Listener
- TCP Sender
- File Reader
- JMS Listener
- Database Reader
- Etc....

Adicionamos um conector 'Fonte' selecionando o tipo de conector necessário a seu caso de uso, disponível na lista suspensa.

Após selecionarmos o conector 'fonte' realizamos a configuração para este conector de Origem. Na imagem temos o Database Reader como tipo de conector selecionado.



Ao selecionarmos o Database Reader como conector fonte, precisamos realizar algumas configurações:

### 1. Poll settings

Intervalo de tempo para que o Mirth realize consultas no banco de dados para obter novos dados. Podemos definir um determinado horário ou executar um trabalho Cron.

### 2. Source settings

Permite que seja definida a ordem de execução das mensagens de origem. Podemos obter uma resposta antes ou após o processamento, definir o número de threads de processamento que serão executados para processar dados.

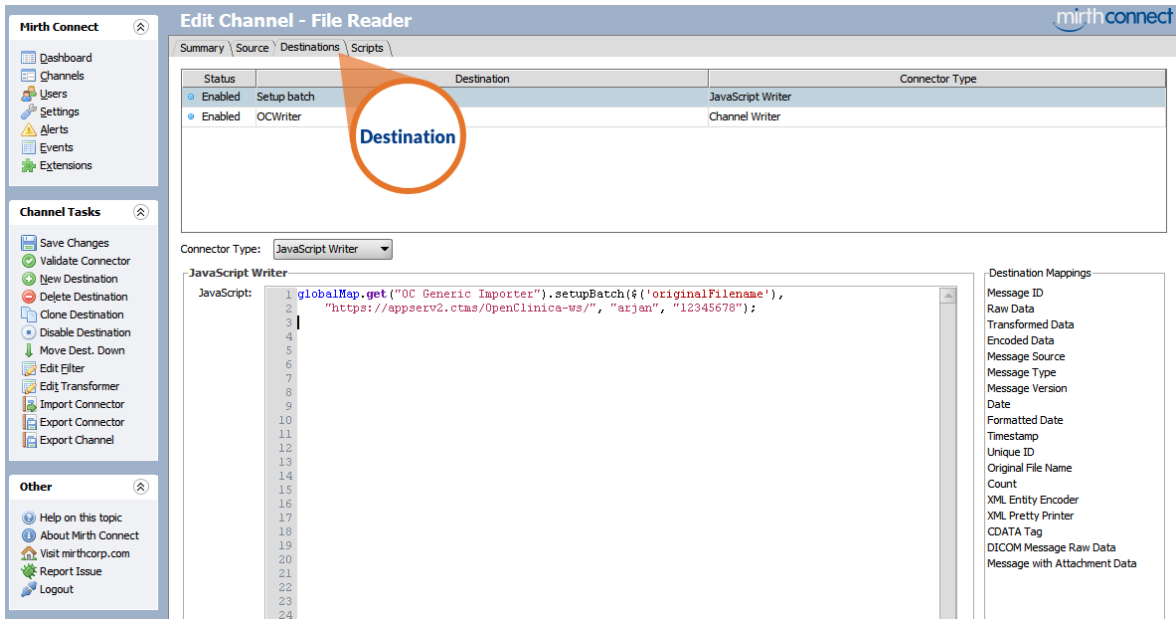
### 3. Database reader settings

As configurações do leitor de banco de dados permitem que o usuário decida que tipo de DB usará: MySQL, Oracle, PostgreSQL, SQLite, etc. Você também pode definir

se deseja salvar os resultados do cache, manter a conexão aberta, quantidade de tentativas em caso de erro.

## Destination

O conector de 'Destino' é bem semelhante ao da 'Fonte', sendo neste, definido o



Status	Setup batch	Destination	Connector Type
Enabled	OCWriter		JavaScript Writer
Enabled			Channel Writer

Connector Type: JavaScript Writer

```

JavaScript:
1 globalMap.get("OC Generic Importer").setupBatch('{{originalFilename'}),
2   "https://appserv2.ctms/OpenClinica-ws/", "arjan", "12345678");
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
    
```

Destination Mappings:

- Message ID
- Raw Data
- Transformed Data
- Encoded Data
- Message Source
- Message Type
- Message Version
- Date
- Formatted Date
- Timestamp
- Unique ID
- Original File Name
- Count
- XML Entity Encoder
- XML Pretty Printer
- CDATA Tag
- DICOM Message Raw Data
- Message with Attachment Data

destino da mensagem. Uma das principais diferenças entre 'Origem' e 'Destino' é que podem haver vários 'destinos' para um Canal.

A barra superior da guia 'Destino' mostra:

- Status
- ID do destino
- Tipo de conector
- 'Cadeia' do canal

A guia 'Destino' tem similaridades com a guia Origem, permitindo igualmente que se selecione o tipo do conector e realizem-se os ajustes das configurações.



## Filters

As guias 'Origem' e 'Destino' dos Canais possuem uma opção de 'Filtros', disponível no painel lateral esquerdo. O filtro permite que você determine se uma mensagem deve ser aceita ou rejeitada.

Ao clicar em 'Filtro' você será direcionado para a tela 'Editar filtro'. Na tela 'Editar filtro', há uma lista vazia dos filtros disponíveis. No final da tela encontramos os parâmetros para os filtros ativos. À direita, existem três guias mostrando as funções de referência, as árvores de mensagens disponíveis e os modelos das mensagens.

## Transformers

Os transformadores estão no lado esquerdo da tela e clicando nele seremos levados para o modo de edição do Transformador. Os transformadores são regras que alteram ou reorganizam os dados na mensagem de origem para criar uma mensagem de destino. Os transformadores basicamente transformam os dados da maneira que precisamos que sejam transmitidos.

## Scripts

Na última guia do Canal estão o (s) Scripts e existem quatro tipos de Scripts a saber:

### 1. Deploy

Este script é executado a cada vez que o mecanismo é iniciado e é usado principalmente para construir e inicializar objetos comuns que precisam serem persistidos enquanto o Mirth está sendo executado.

### 2. Shutdown

Este script é executado todas as vezes em que o mecanismo é interrompido e é usado para limpar tudo o que foi criado pelo script de Deploy.

### 3. Preprocessor

Este script é executado imediatamente quando do recebimento de uma mensagem e antes que a mensagem seja normalizada para XML. O Mirth trabalha internamente com o formato XML.

#### 4. Postprocessor

Este script é executado imediatamente após a mensagem ser enviada e contém variáveis específicas para o canal.

### **Bem, estes são os elementos básicos da interface do Mirth**

Como o Mirth é um sistema grande, não é possível realizarmos uma análise aprofundada em todos os elementos do Mirth Connect em um único artigo, mas tentamos cobrir todos os detalhes do "Canal", neste.

Se você se interessou e quer saber mais sobre o Mirth Connect, a InterOpera disponibiliza um curso ON-LINE com 03 aulas ao vivo cobrindo na prática as principais funcionalidades do software.

Caso sua organização tenha pressa em habilitar o intercâmbio de dados, você pode se interessar por uma instância Mirth Connect localmente ou em uma nuvem segura como a AWS T2.Large.

Conte com os serviços de Consultoria, Engenharia, Treinamentos, Deploy, Desenvolvimento e Governança da InteroperaHIT.

Saiba mais

[interopera.com.br](http://interopera.com.br)

[info@interopera.com.br](mailto:info@interopera.com.br)

**Paulo R. Rades, cpTICS, HL7**