

## MENSAGENS DE ACK

As mensagens HL7 de confirmação, ou ACK, são fundamentais para garantir que a comunicação HL7 em andamento seja contínua e sem problemas.

O conceito de [ACK](#) é comumente usado em muitos protocolos de rede de dados, incluindo o TCP, então você provavelmente está familiarizado com ele.

Embora seja conceitualmente simples (o sistema de recepção reconhece uma mensagem específica enviada pelo do sistema de envio), há algumas variações como: modo avançado, modo original, com limitação de taxa...

### A estrutura de uma mensagem ACK

Vamos fazer uma rápida visão em como uma mensagem ACK é gerada.

Digamos que uma mensagem HL7 de ADT (Admit, Discharge, Transfer) para admissão foi recebida com o seguinte MSH (cabeçalho da mensagem):

```
MSH|^~\&|EPICADT|DH|LABADT|DH|201701011226||ADT^A01|
HL7MSG00001|P|2.3|
```

Se a mensagem foi reconhecida e aceita, então a mensagem ACK de resposta (conforme o modo original) será assim:

```
MSH|^~\&|LABADT|DH|EPICADT|DH|201701011228||ACK^A01^ACK|
|HL7ACK00001|P|2.3|
MSA|AA|HL7MSG00001
```

Observe que:

1. Os sistemas de origem e de destino foram comutados;
2. O segmento MSA contém o valor "AA" que indica aceitação; e
3. O segmento MSA também contém o ID da mensagem original

Parece simples, mas como você vai ver, as regras utilizadas para chegar a esta mensagem podem ser bastante complicadas.

## A necessidade das mensagens de ACK

A necessidade das mensagens de reconhecimento fica melhor compreendida quando entendemos que:

1. O HL7 foi projetado sobre o conceito de eventos de disparo (trigger events) - ou seja, alguém foi admitido, uma solicitação de exames de laboratório tem de ser realizado, os resultados do exame laboratorial foram recebidos, o paciente tem que ser reencaminhado, etc.
2. Os Sistemas de TI em saúde são muito "especializados" - existem sistemas (muitas vezes de diferentes fornecedores) que se concentram apenas em ADT (admissão, altas e transferências), gestão de práticas (agendamento e faturamento), laboratórios (ordens de laboratório, Etc)

**Vejamos:** se um evento acontece em um sistema, como a admissão de um paciente), esse evento deve ser enviado para os demais sistemas (por exemplo para o sistema do laboratório) e enviará informações como os identificadores internos do paciente (caso contrário, como o laboratório saberá se a solicitação do exame é para um paciente válido ou não, qual identificador usar etc.).

**Observe que essas mensagens geralmente não são solicitadas - ou seja, a mensagem de ADT é enviada para todos os sistemas interessados, porém nenhum dos sistemas envolvidos a solicitou.**

Além disso o volume de mensagens recebidas por esses sistemas pode ser grande, daí a possibilidade de a mensagem não ter sido recebida ou processada. O ACK então servirá como confirmação de que:

1. A mensagem (especificada por um identificador) foi recebida;
2. A mensagem é válida com base nas regras de processamento HL7 (veremos mais sobre isso em um minuto); e opcionalmente
3. Se os dados da mensagem foram entregues em um armazenamento temporário, como uma fila para processamento, ou em armazenamento permanente, como um banco de dados (mais sobre isso em um minuto também).

## Os tipos de mensagens ACK e as regras para seu processamento

Como você pode ver, a mensagem de ACK **não** é como o reconhecimento de entrega que você recebe quando você envia um e-mail ou uma mensagem de texto – O ACK não é uma mensagem de "eu tenho isso".

É possível especificar se as regras de processamento originais ou avançadas devem ser aplicadas à mensagem. Com base nessa especificação, a mensagem de entrada é processada de forma diferente, e um tipo diferente de mensagem de ACK é enviada ao sistema emissor.

A mensagem ACK e as suas regras de processamento são definidas com base no conteúdo do segmento MSH (cabeçalho da mensagem) (os detalhes sobre o segmento MSH foram discutidos em **outro artigo**).

### O Modo Original

O processamento no modo original é indicado através dos campos 15 e 16 do segmento MSH da mensagem de entrada se estes estiverem **nulos ou vazios**.

Qualquer mensagem de entrada com um segmento MSH indicando o processamento do modo original validará a correta sintaxe e passará por um processo em duas etapas:

#### **PASSO 1 - Validação de protocolos**

Usado para garantir que:

1. O valor no campo do tipo da mensagem (MSH - campo # 9) seja aceitável pelo receptor, ou seja, uma mensagem ADT^A20 pode ser rejeitada pelo sistema de faturamento;
2. O valor no campo ID da versão (MSH-12) seja aceitável pelo receptor, ou seja, se os sistemas esperam HL7 v2.3.1 e o campo diz v2.6, então, teremos Falha de Comunicação

3. O valor do ID de processamento (MSH-11) seja apropriado para o processo da aplicação que trata a mensagem.

Se qualquer uma dessas verificações falhar, o protocolo do software rejeitará a mensagem através de uma mensagem ACK contendo "AR" no campo do código de confirmação (MSA-1). Se a verificação não falhar, a mensagem é transferida para a aplicação.

## PASSO 2 - Validação de aplicações

As verificações de validação do aplicativo são:

1. Se a aplicação processar a mensagem com êxito, irá gerar a mensagem de resposta funcional com um valor de AA no código de confirmação (MSA-1);
2. Ou o aplicativo enviará uma resposta de erro, com um valor de AE no código de confirmação (MSA-1);
3. Se o aplicativo falhar ao processar (rejeitar) a mensagem por causa do tempo de atividade do sistema, ou por outros motivos que não por erros de formatação ou validação, a mensagem de resposta conterá o valor "AR" no código de confirmação (MSA-1);

Às vezes, uma mensagem poderá ser reenviada mais tarde, mas isso depende do sistema de envio e se a implementação permitirem.

## Modo avançado

O processamento no modo avançado é indicado se pelo menos um dos campos 15 e 16 do segmento MSH da mensagem de entrada **não for nulo**.

O modo avançado exige que a aplicação receptora assuma uma responsabilidade adicional, nomeadamente que:

1. A mensagem de entrada seja recebida e armazenada;
2. A mensagem passa pela validação de sintaxe;
3. O tipo de mensagem (ADT-A19 pode ser processado), a versão está correta (HL7 v2.3.1 e não v2.6 por exemplo) e o ID de processamento está correto, e em caso negativo, uma mensagem de rejeição de Commit Reject (CR), será enviada

Com base nestas regras, o sistema receptor enviará:

1. Um commit accept (CA) se tudo estiver OK;
2. Um commit reject (CR) como acima; ou
3. Uma confirmação de erro (CE), para qualquer outro tipo de erro.

## O ACK Customizado

E o que seria da TI em Saúde sem alguma personalização? E não surpreendentemente, é possível enviar um ACK não-HL7/Static String.

Isto é uma personalização da mensagem de reconhecimento e é simplesmente uma sequência de caracteres de texto (em vez de um ACK no formato do HL7).

Esses tipos de ACKs são usados quando um sistema emissor é incapaz de receber mensagens formatadas em HL7 ou criá-las.

## Limitação de taxa com mensagens ACK

O padrão HL7 define que os sistemas emissores não podem enviar uma outra mensagem para um sistema até que tenha recebido um ACK em resposta.

Isto na verdade não é correto. Não faz parte da especificação do HL7.

Geralmente a forma como os sistemas com HL7 são implementados na *prática* é que devem garantir que as mensagens sejam tratadas apropriadamente.

Isto feito, presume-se, que para garantir que se as mensagens forem rejeitadas devido a erros de conteúdo, formatos de mensagem, tempo de inatividade do

sistema etc., elas possam ser corrigidas na origem ou na fila ou até que o sistema de destino volte a funcionar.

Observamos que se a próxima mensagem não for enviada até que o ACK seja recebido, é possível um delay no envio de mensagens de entrada, por esta falta de retorno do ACK.

Explicamos em maiores detalhes:

Até a v2.7, os únicos ACKs "oficiais" eram o clássico Modo Original e o Modo Avançado. Não havia nenhuma outra maneira para que um sistema de upstream realmente dentro do padrão, soubesse o que você estava fazendo com a mensagem. A suposição válida seria:

- A mensagem foi processada sem erro após receber o ACK ou;
- A mensagem não foi processada corretamente através de um NACK ou no ACK.

Isso faz sentido, pois seria perigoso enviar / processar a atualização de um registro de paciente se você não foi capaz de processar o evento inicial que deu origem a esse paciente em seu sistema, ou receber uma nota de atualização para um paciente que não existe, etc).

A maioria do design do HL7 é linear e não possui um conceito consistente, onde sempre houveram exceções sobre isto (principalmente os sistemas que não tinham capacidade de responder com um ACK).

Na v2.7, um novo campo foi adicionado ao segmento MSH do ACK, o 15º, que permite ao sistema receptor indicar ao sistema de envio o que fazer com a mensagem depois de recebê-la.

Enquanto isto aconteceu durante anos, a intenção deste projeto era acomodar os motores de integração, pois estes podem receber uma mensagem e em seguida, intercambia-la para uma multidão de sistemas e assim fornecer para o sistema de envio, alguma informação sobre como os sistemas receptores a processaram.

Ficou claro nos projetos de motores de integração (e as equipes destes continuam com este trabalho), que estes são responsáveis pela solução de problemas entre os sistemas de ADT ou EHR que gerou o tipo de mensagem original.

Se você precisa integrar os dados de seu EHR com outras aplicações sem se tornar um especialista em HL7, a Interopera pode ajudar.

Conheça os nossos [Serviços de Integração](#) e nossos [Treinamentos em HL7](#).