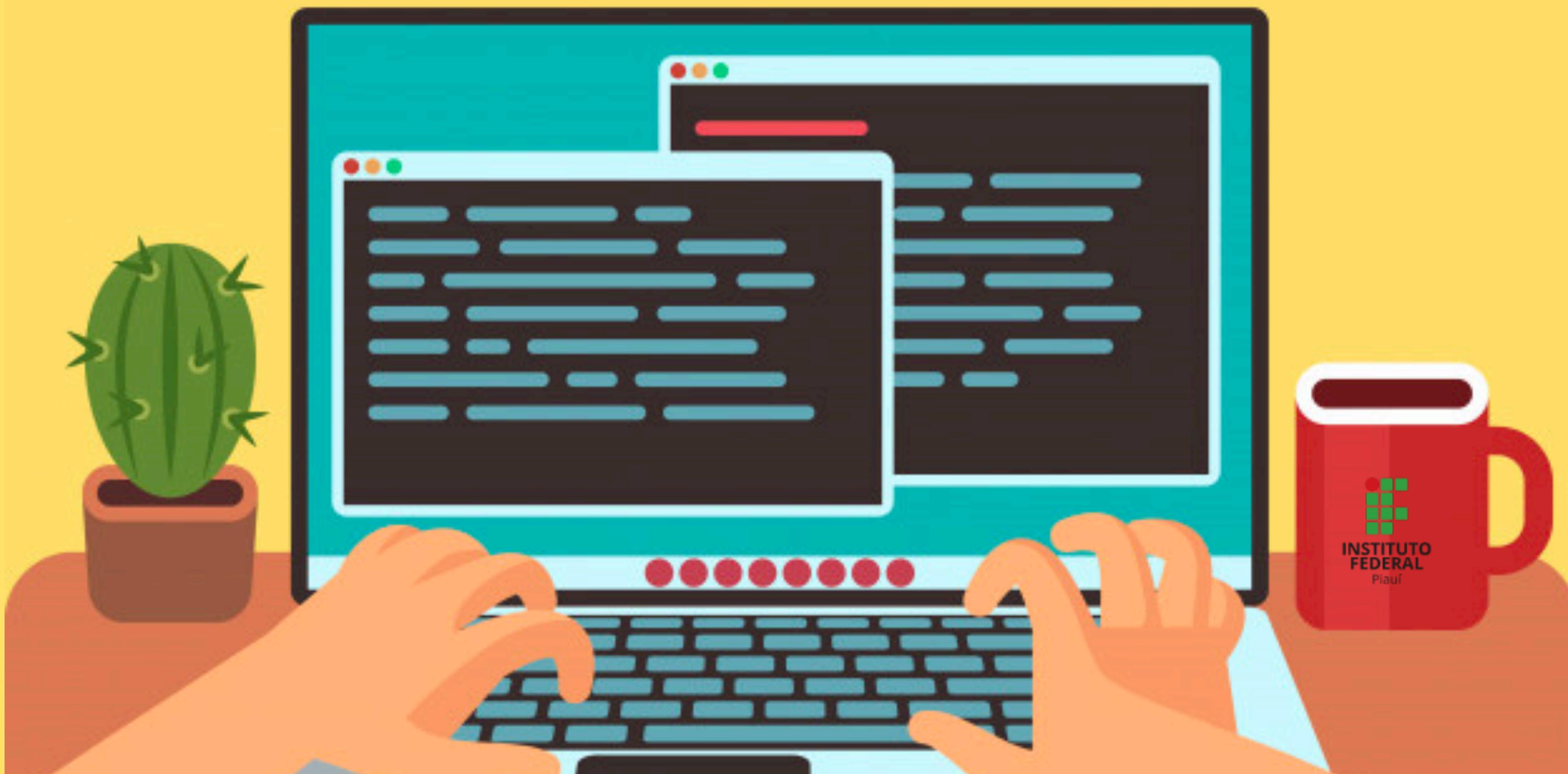
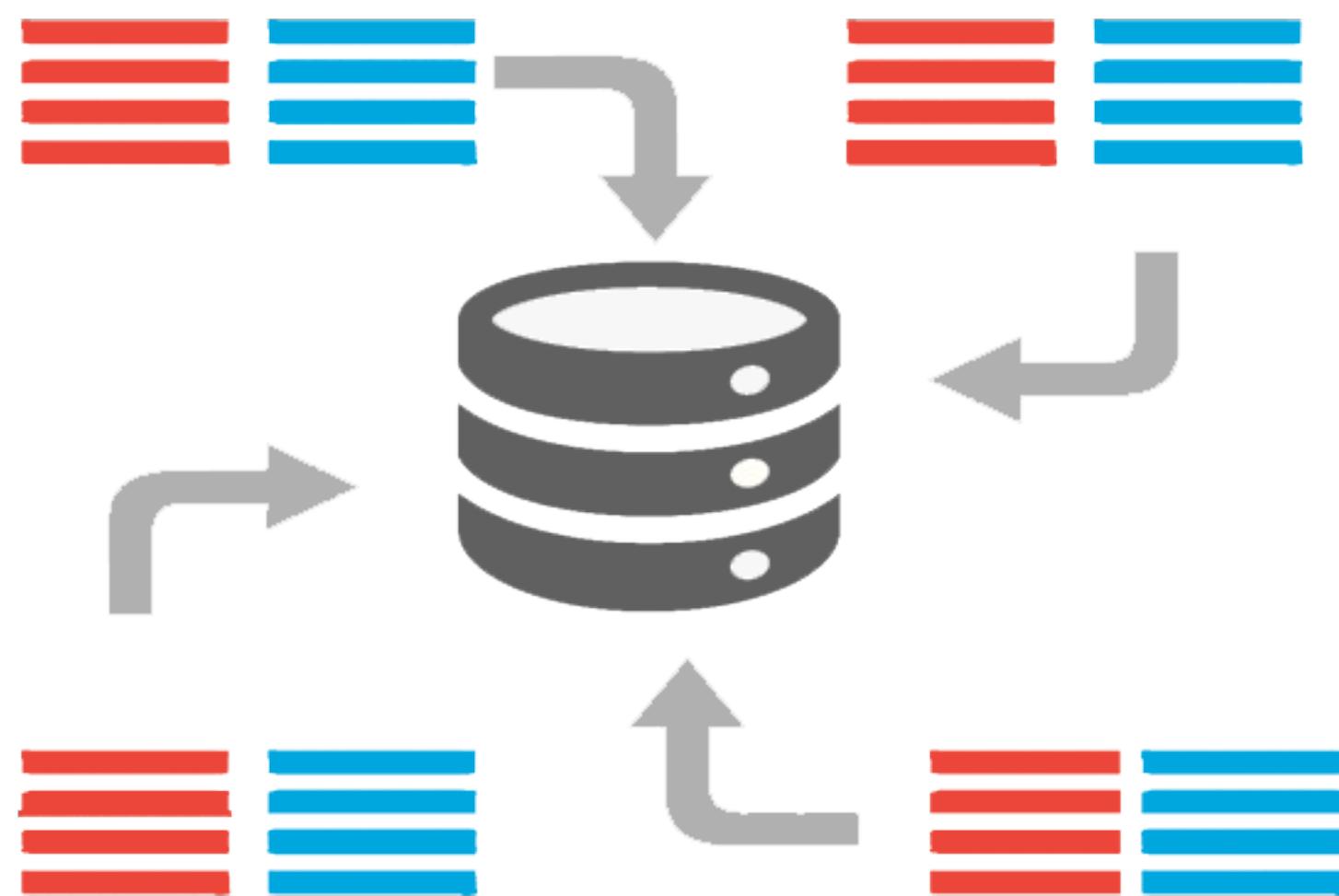


REGRAS ATIVAS NO POSTGRESQL

Prof. Rangel Nunes

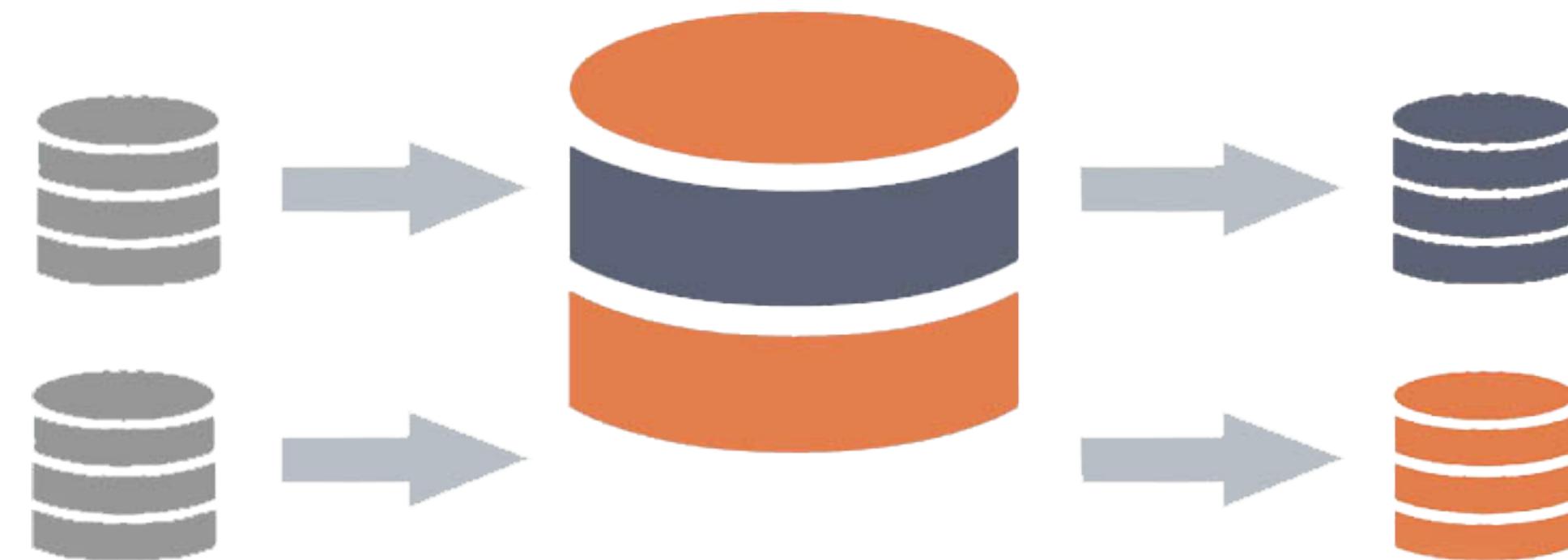


SGBD PASSIVO



- ✓ Os SGBDs fornecem capacidade de armazenamento de grande quantidade de informações
- ✓ Além disso, dispõem de ferramentas interativas para manipulação e atualização de dados
- ✗ Porém, suas principais operações ocorrem devido a requisições da aplicação ou dos usuários

SGBD ATIVO



- ✖ Os sistemas passivos passaram a não mais atender às necessidades
- ✓ Um SGBDA estende as funcionalidades de um SGBD passivo
- ✓ Tornando-o capaz de responder a eventos gerados interna ou externamente ao sistema

CARACTERÍSTICAS DO SGBDA

- Em setembro de 1996 foi publicado o Manifesto dos SGBDAs*
- Ele descreveu algumas características essenciais para que um determinado SGBD seja considerado um SGBDA
- São elas:
 - ✓ Um SGBDA é um sistema gerenciador de BD
 - ✓ Um SGBDA suporta a definição e o gerenciamento de regras ECA (Evento-Condição-Ação)
 - ✓ Um SGBDA possui um modelo de execução

MODELO E.C.A

Evento	É o responsável por ativar uma regra. Esses eventos são normalmente operações de atualização de banco de dados aplicadas explicitamente
Condição	Elá determina se a ação da regra deve ser executada. Uma vez ocorrido o evento, uma condição opcional pode ser avaliada. Se nenhuma condição for especificada, a ação será disparada pelo evento. Se uma condição for especificada, primeiramente ela será avaliada, e somente se seu resultado for verdadeiro a ação da regra será executada
Ação	Normalmente é uma sucessão de declarações SQL, mas também poderia ser uma transação de BD ou um programa externo que será executado automaticamente



E O QUE EU POSSO FAZER COM ELE?

- Criar notificações sobre certos acontecimentos que ocorrem internamente
- Controle de restrições de integridade
- Auditoria de sistemas de informação

COM ISSO... SURGIRAM AS TRIGGERS

- **Trigger** ou gatilho é um procedimento SQL armazenado em um banco de dados que depende de um evento (INSERT, DELETE ou UPDATE) para que seja disparado
- Uma característica é que ele pode ser disparado antes (BEFORE) ou depois (AFTER) da ocorrência de um evento

SINTAXE DE CRIAÇÃO DE TRIGGER , NO POSTGRESQL

```
CREATE TRIGGER nome {BEFORE | AFTER} {evento [OR ...]}  
ON tabela [FOR [ EACH ] {ROW | STATEMENT}]  
EXECUTE PROCEDURE nome_da_função (argumentos);
```



POR ISSO, AS FUNCTIONS SÃO TÃO IMPORTANTES...

```
CREATE OR REPLACE FUNCTION nome_da_funcao() RETURNS TRIGGER AS
$$
BEGIN
    (comandos)
END;
$$ LANGUAGE plpgsql;
```



TEM MAIS REGRAS ATIVAS. . .

- No PostgreSQL, além das Triggers, temos o conceito de **RULES**
- Elas também servem para criar uma regra aplicada à tabela ou a uma visão
- Apesar de mais simples, nos dão funcionalidade que não podemos fazer com TRIGGERS

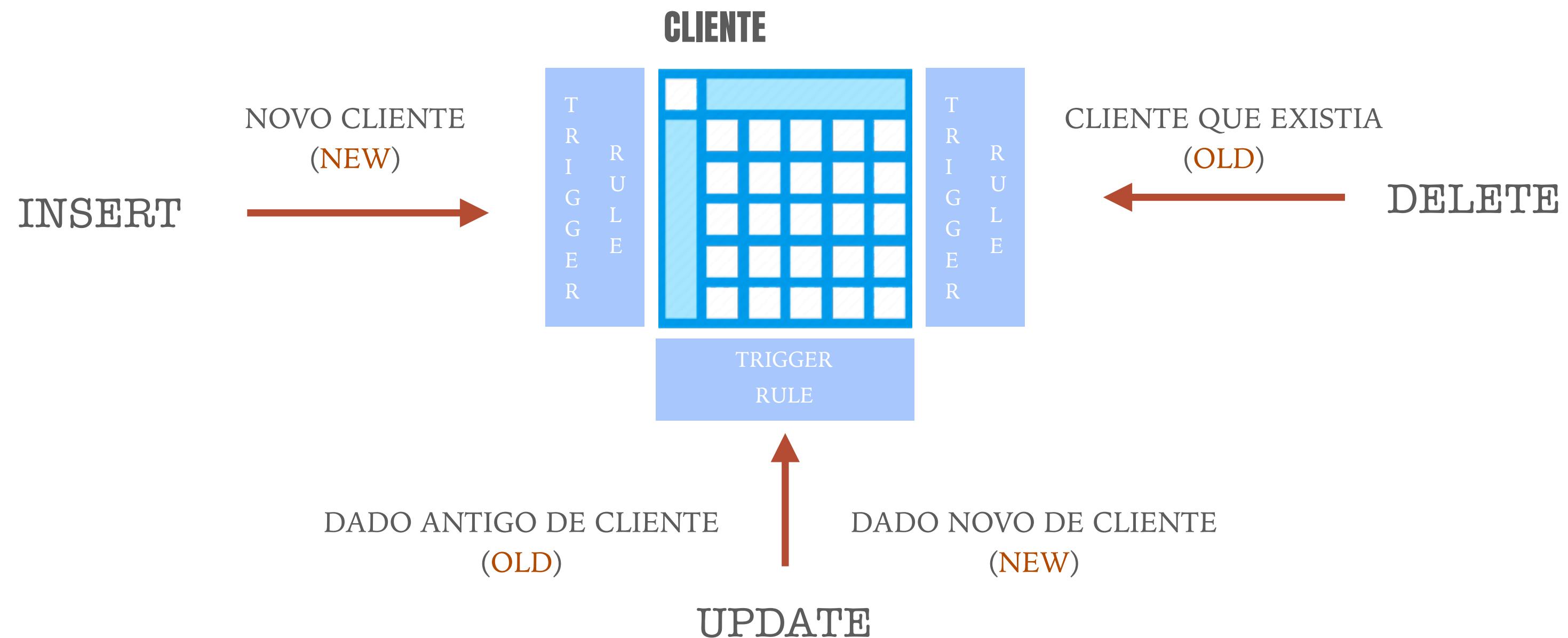
SINTAXE DE CRIAÇÃO DE RULES, NO POSTGRESQL

```
CREATE [OR REPLACE] RULE nome AS ON evento
TO tabela [WHERE condição]
DO [ALSO | INSTEAD] {NOTHING | comando | (comando; comando ...)}
```



COM AS REGRAS ATIVAS, TEMOS ACESSO ÀS VARIÁVEIS NEW E OLD

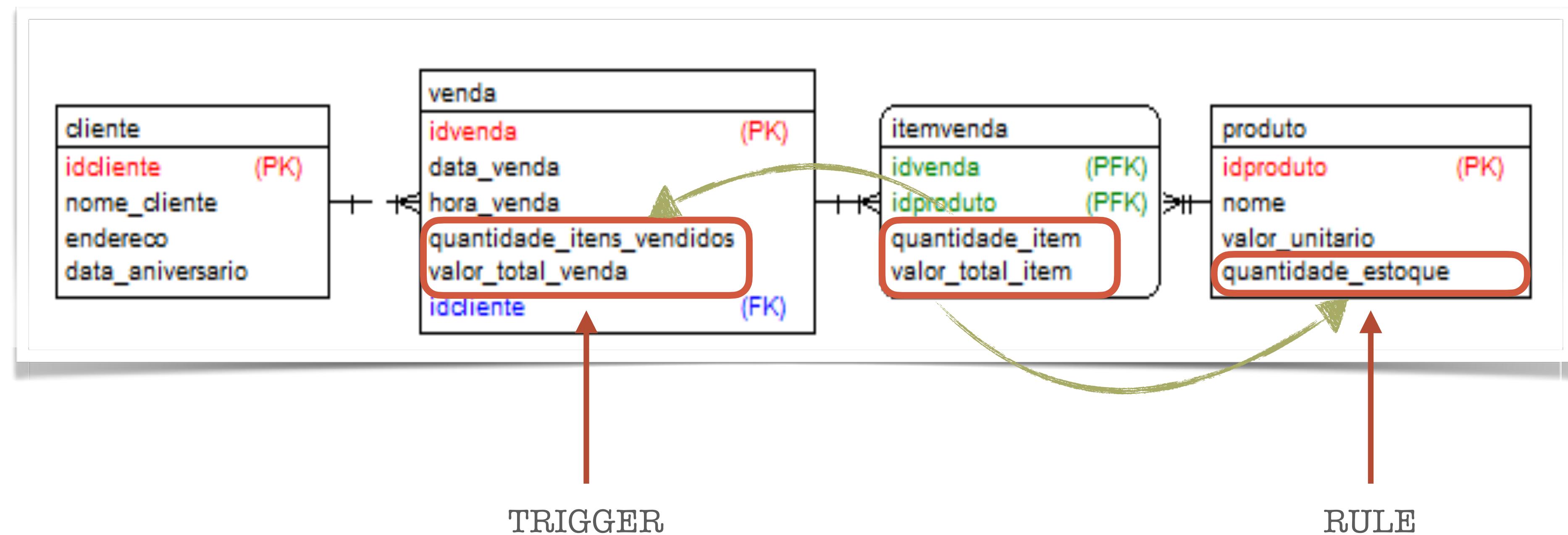
- Onde **NEW** representa o novo conteúdo para cada uma das colunas
- **OLD** representa o conteúdo anterior



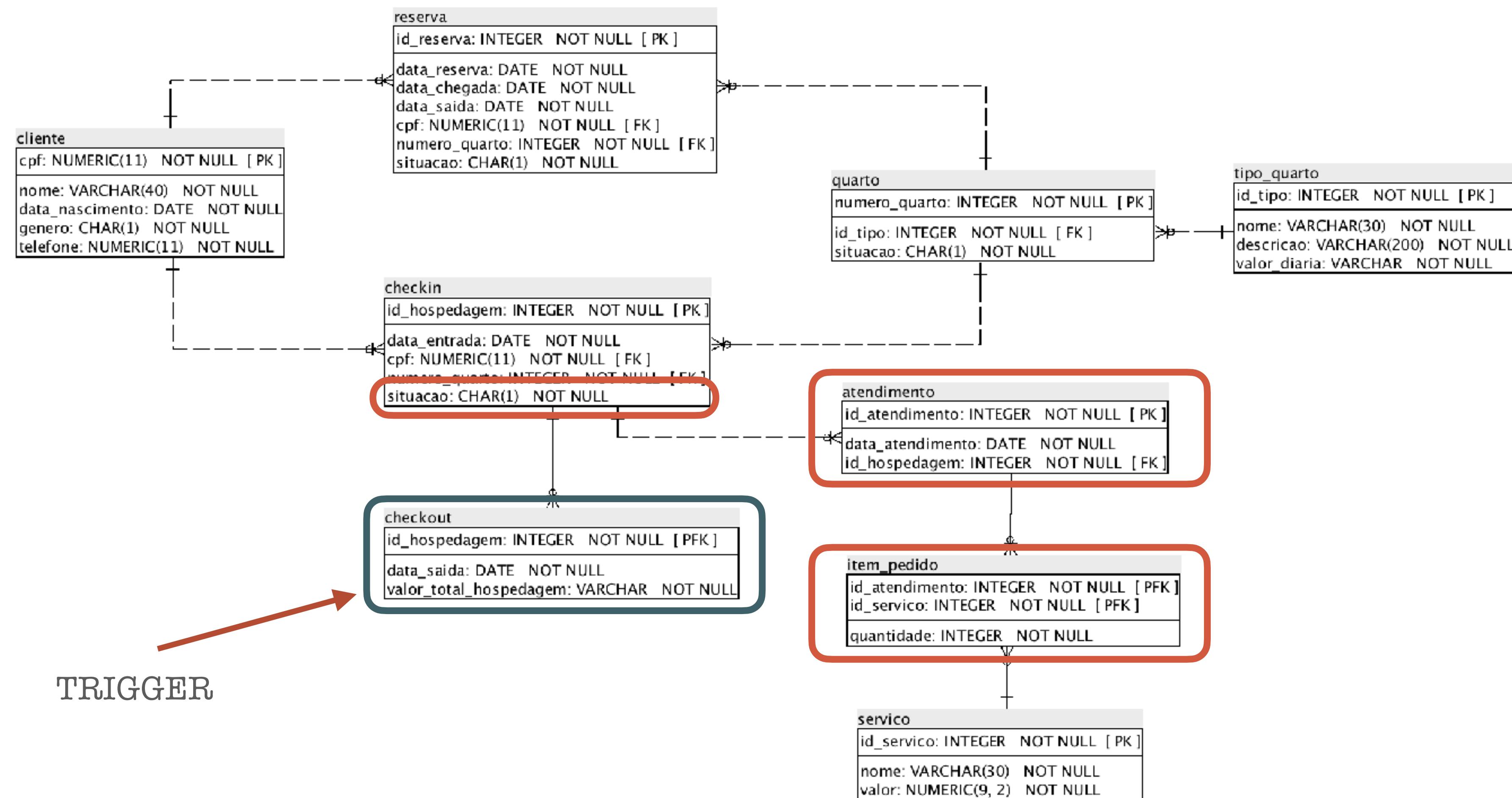
COMO IMPLEMENTAR REGRAS ATIVAS?



ESTUDO DE CASO



ATIVIDADE



ATIVIDADE

- Deve-se verificar quantos dias o hóspede ficou no hotel, levando-se em consideração o tipo de quarto dele;
- Verificar se durante a sua hospedagem, ele pediu algum serviço;