

s) A integridade dos dados na base de dados será assegurada através dos seguintes mecanismos:

Consistência de atributos derivados:

- **PRODUTO.Quant_Em_Maquinas:** Este campo, na tabela PRODUTO, deve refletir a soma das quantidades em stock de um produto em todas as máquinas (i.e., a soma de POSSUI.Quant_Em_Stock).
- **PRODUTO.Quant_Em_Stock:** Este campo, na tabela PRODUTO, deve corresponder à soma das quantidades em stock de um produto em todos os armazéns (i.e., a soma de TEM.Quant_Em_Stock).
- **POSSUI.Quant_Em_Stock:** Este campo, na tabela POSSUI, deve refletir a soma das QUANTIDADE_ATUAL de um produto em todos os compartimentos (COMPARTIMENTO.Quantidade_Atual) que contêm esse produto na máquina específica.
- **POSSUI.Capacidade_Max:** Este campo, na tabela POSSUI, deve refletir a soma das CAPACIDADE_MAX de um produto em todos os compartimentos (COMPARTIMENTO.Capacidade_Max) que contêm esse produto na máquina específica.
- **Mecanismo:** A utilização de *triggers* AFTER INSERT OR UPDATE OR DELETE nas tabelas POSSUI, TEM e COMPARTIMENTO permitirá recalcular e atualizar os campos correspondentes nas tabelas PRODUTO e POSSUI.

Restrições de valor e intervalos:

- **Valores Não Negativos:** Atributos como PRECO_MEDIO (PRODUTO), QUANTIDADE_ATUAL, CAPACIDADE_MAX, QUANTIDADE_MINIMA (COMPARTIMENTO), CAPACIDADE_CARGA, AUTONOMIA_KM (VEICULO), DISTANCIA_KM (VIAGEM, ROTA), QUANTIDADE_REPOSTA (REPOSICAO), UNIDADES (TRANSPORTA) e QUANTIDADE (PEDE) devem ser validados para garantir que são sempre positivos ou zero, se aplicável.
- **Mecanismo:** *Triggers* BEFORE INSERT OR UPDATE serão utilizados para validar os valores antes da operação e lançar exceções se forem inválidos. Procedimentos e funções que manipulem estes valores deverão incluir validação de entrada.
- **Datas Válidas:**
 - **VIAGEM:** A DATA_HORA_INICIO deve ser estritamente anterior à DATA_HORA_FIM.
 - **DATA_HORA(VENDA), DATA_REPOSICAO, DATA_MANUTENCAO, DATA_RECLAMACAO, DATA_ENCOMENDA, DATA_VISITA:** Estas datas não devem ser futuras (i.e., devem ser SYSDATE ou anteriores).
 - **DATA_ENCOMENDA de Produtos:** A data de início de avaliação para a encomenda deve ser anterior à data atual.
- **Mecanismo:** *Triggers* BEFORE INSERT OR UPDATE serão usados para validar os intervalos e a temporalidade das datas. Procedimentos e funções que recebam datas como parâmetros deverão incluir validação explícita.

Regras de negócio específicas e complexas:

- **Venda de Produto (Stock Suficiente e Atualização):**
 - Uma venda só pode ser registada se houver stock suficiente do produto no compartimento específico da máquina.
 - Após a venda, o stock deve ser automaticamente decrementado no COMPARTIMENTO, na tabela POSSUI e na tabela PRODUTO (Quant_Em_Maquinas). O número de UNIDADES_VENDIDAS no PRODUTO deve ser incrementado.
 - Se, após a venda, a máquina ficar completamente vazia, sem stock de nenhum produto, o seu STATUS (na tabela MAQUINA) deve ser atualizado para 'SEM STOCK'.
 - **Mecanismo:** Um *trigger* BEFORE INSERT na tabela VENDA será utilizado para verificar o stock antes da venda e lançar uma exceção se for insuficiente. Um *trigger* AFTER INSERT na tabela VENDA realizará os decrementos de stock e as atualizações de status da máquina.
- **Abastecimento de Produto (Validação de Quantidade e Atualização de Stock):**
 - Ao registar uma reposição (abastecimento), a QUANTIDADE_REPOSTA não pode ser negativa, nem pode fazer com que a quantidade atual exceda a CAPACIDADE_MAX do compartimento.
 - Após a reposição, o stock deve ser incrementado no COMPARTIMENTO, na tabela POSSUI e na tabela PRODUTO (Quant_Em_Maquinas). A DATA_HORA_ULTIMA_VISITA da máquina pode ser atualizada.
 - **Mecanismo:** Um *trigger* BEFORE INSERT OR UPDATE na tabela REPOSICAO será utilizado para validar a QUANTIDADE_REPOSTA. Um *trigger* AFTER INSERT na tabela REPOSICAO realizará os incrementos de stock e a atualização da DATA_HORA_ULTIMA_VISITA da máquina.
- **Atualização de Quantidade Transportada em VIAGEM:**
 - Quando um produto é abastecido e está associado a uma VIAGEM (através de uma VISITA), a quantidade transportada desse produto para essa VIAGEM deve ser atualizada, em conformidade com a quantidade abastecida.
 - **Mecanismo:** Um *trigger* AFTER INSERT na tabela REPOSICAO que, através da VISITA associada, localize a VIAGEM e atualize a quantidade na tabela TRANSPORTA.
- **Operações em VIAGENS de Abastecimento (Estado de Início):**
 - Não devem ser permitidas operações que dependam de uma VIAGEM de abastecimento se esta ainda não tiver sido iniciada.
 - **Mecanismo:** Procedimentos ou funções que manipulem VIAGENS deverão incluir uma verificação da DATA_HORA_INICIO da VIAGEM, comparando-a com SYSDATE.
- **Máquinas Offline:**
 - Se, num intervalo de 30 minutos, não for recebida nenhuma informação de uma máquina, esta é considerada offline e deve ser visitada o mais rapidamente possível.
 - **Mecanismo:** Um procedimento agendado que verifique o DATA_HORA_ULTIMO_REPORT das máquinas na tabela MAQUINA e atualize o STATUS da máquina para 'OFFLINE' se o último reporte for mais antigo do que 30 minutos.

t) Para determinar o impacto físico e estimar o espaço ocupado pelas tabelas de maior volume de dados no sistema, consideram-se as tabelas VENDA, REPOSICAO, ENVIA_EVENTO, VISITA e TRANSPORTA. Estas tabelas registam transações e eventos de alta frequência, esperando-se um acumular significativo de registos ao longo do tempo.

A estimativa do espaço será baseada no tamanho do bloco de dados do Oracle (4KB, ou 4096 bytes) e no parâmetro PCTFREE.

A resolução desta alínea foi inspirada na resolução do exercício 9) a) do exame 202122_AABD_Normal_v1_01 feita em aula.

1. Determinação do Tamanho Médio de uma Linha:

O cálculo do tamanho médio de um registo para cada tabela envolve a soma do tamanho dos seus atributos, **acrescido de um cabeçalho de linha fixo de 5 bytes**. Para os tipos de dados:

- NUMBER: 4 bytes.
- DATE: 7 bytes.
- VARCHAR2(n): n/2 bytes.

Cálculo de tamanho por linha:

- **VENDA:**
 - Cabeçalho de linha: 5 bytes
 - ID_VENDA (NUMBER): 4 bytes
 - ID_MAQUINA (NUMBER): 4 bytes
 - ID_PRODUTO (NUMBER): 4 bytes
 - DATA_HORA (DATE): 7 bytes
 - QUANTIDADE (NUMBER): 4 bytes
 - **Tamanho médio da linha de VENDA:** $5 + 4 + 4 + 4 + 7 + 4 = 28$ bytes
- **ENVIA_EVENTO:**
 - Cabeçalho de linha: 5 bytes
 - ID_RECLAMACAO (NUMBER): 4 bytes
 - ID_EVENTO (NUMBER): 4 bytes
 - DATA_HORA_ENVIO (DATE): 7 bytes
 - **Tamanho médio da linha de ENVIA_EVENTO:** $5 + 4 + 4 + 7 = 20$ bytes
- **REPOSICAO:**
 - Cabeçalho de linha: 5 bytes
 - ID_REPOSICAO (NUMBER): 4 bytes
 - ID_COMPARTIMENTO (NUMBER): 4 bytes
 - ID_PRODUTO (NUMBER): 4 bytes
 - ID_VISITA (NUMBER): 4 bytes
 - QUANTIDADE_REPOSTA (NUMBER): 4 bytes
 - DATA_REPOSICAO (DATE): 7 bytes
 - DESCRICAO (VARCHAR2(50) - média de 25 caracteres): 25 bytes
 - **Tamanho médio da linha de REPOSICAO:** $5 + 4 + 4 + 4 + 4 + 4 + 7 + 25 = 57$ bytes

- **VISITA:**
 - Cabeçalho de linha: 5 bytes
 - ID_VISITA (NUMBER): 4 bytes
 - ID_VIAGEM (NUMBER): 4 bytes
 - ID_RECLAMACAO (NUMBER): 4 bytes
 - ID_MAQUINA (NUMBER): 4 bytes
 - MOTIVO (VARCHAR2(50) - média de 25 caracteres): 25 bytes
 - DISTANCIA (NUMBER): 4 bytes
 - DATA_VISITA (DATE): 7 bytes
 - **Tamanho médio da linha de VISITA:** $5 + 4 + 4 + 4 + 4 + 25 + 4 + 7 = 57$ bytes
- **TRANSPORTA:**
 - Cabeçalho de linha: 5 bytes
 - ID_VIAGEM (NUMBER): 4 bytes
 - ID_PRODUTO (NUMBER): 4 bytes
 - UNIDADES (NUMBER): 4 bytes
 - **Tamanho médio da linha de TRANSPORTA:** $5 + 4 + 4 + 4 = 17$ bytes

2. Cálculo do Número de Registos por Bloco:

Um bloco de dados Oracle tem um *overhead* fixo considerado de 84 bytes (cabeçalho do bloco). O PCTFREE define a percentagem de espaço que o Oracle reserva em cada bloco para futuras atualizações de linhas existentes. Considera-se o tamanho do bloco de dados como 4KB (4096 bytes).

- **Tamanho do Bloco de Dados:** 4096 bytes
- **Overhead Fixo do Bloco:** 84 bytes
- **Espaço útil do bloco (sem PCTFREE):** $4096 - 84 = 4012$ bytes
- **PCTFREE:** Para todas as tabelas consideradas (VENDA, REPOSICAO, ENVIA_EVENTO, VISITA, TRANSPORTA), o PCTFREE será definido como **0%**. Esta escolha justifica-se pelo facto de estas tabelas serem predominantemente de inserção, com poucas ou nenhuma atualizações que resultem no crescimento do tamanho das linhas existentes.
 - **Espaço disponível para novos dados (com PCTFREE = 0%):** 4012 bytes
- **Número de Registos por Bloco:** Espaço disponível para novos dados / Tamanho Médio da Linha
 - **VENDA:** $4012 / 28 = \sim 143$ registos por bloco
 - **ENVIA_EVENTO:** $4012 / 20 = \sim 200$ registos por bloco
 - **REPOSICAO:** $4012 / 57 = \sim 70$ registos por bloco
 - **VISITA:** $4012 / 57 = \sim 70$ registos por bloco
 - **TRANSPORTA:** $4012 / 17 = \sim 235$ registos por bloco

3. Estimativa do Crescimento Anual de Registos:

Para os cálculos do número de registos por ano foram utilizados valores não reais nem estimados, apenas exemplos:

- **VENDA:**

- Considerando 100 máquinas ativas, com uma média de 50 vendas por máquina por dia.
- $100 \text{ máquinas} * 50 \text{ vendas/dia} * 365 \text{ dias/ano} = 1.825.000 \text{ registros/ano.}$
- **ENVIA_EVENTO:**
 - Considerando relatórios a cada 10 minutos (144 por dia) por máquina.
 - $100 \text{ máquinas} * 144 \text{ eventos/dia} * 365 \text{ dias/ano} = 5.256.000 \text{ registros/ano.}$
- **REPOSICAO:**
 - Considerando 1 reposição por máquina por semana, com 10 produtos repostos por reposição.
 - $100 \text{ máquinas} * 52 \text{ semanas/ano} * 10 \text{ produtos/reposição} = 52.000 \text{ registros/ano.}$
- **VISITA:**
 - Considerando 1 visita por máquina por semana.
 - $100 \text{ máquinas} * 52 \text{ semanas/ano} = 5.200 \text{ registros/ano.}$
- **TRANSPORTA:**
 - Considerando que cada visita de reposição está associada a itens transportados (10 produtos por visita).
 - $5.200 \text{ visitas/ano} * 10 \text{ produtos/visita} = 52.000 \text{ registros/ano.}$

4. Cálculo do Espaço Total (INITIAL) Anual:

O espaço anual necessário para cada tabela é calculado multiplicando o número de blocos necessários pelo tamanho de cada bloco (neste caso, 4096 bytes).

- **Venda**
 - Número de blocos necessários = registros anuais / registros por bloco = $1.825.000 / 143 \approx 12.763 \text{ blocos}$
 - Espaço anual = $12.763 \text{ blocos} \times 4096 \text{ bytes/bloco} \approx \mathbf{52,3 \text{ MB}}$
- **Envia_Evento**
 - Número de blocos necessários = $5.256.000 / 200 \approx 26.280 \text{ blocos}$
 - Espaço anual = $26.280 \text{ blocos} \times 4096 \text{ bytes/bloco} \approx \mathbf{107,6 \text{ MB}}$
- **Reposição**
 - Número de blocos necessários = $52.000 / 70 \approx 743 \text{ blocos}$
 - Espaço anual = $743 \text{ blocos} \times 4096 \text{ bytes/bloco} \approx \mathbf{3,0 \text{ MB}}$
- **Visita**
 - Número de blocos necessários = $5.200 / 70 \approx 75 \text{ blocos}$
 - Espaço anual = $75 \text{ blocos} \times 4096 \text{ bytes/bloco} \approx \mathbf{0,3 \text{ MB}}$
- **Transporta**
 - Número de blocos necessários = $52.000 / 235 \approx 222 \text{ blocos}$
 - Espaço anual = $222 \text{ blocos} \times 4096 \text{ bytes/bloco} \approx \mathbf{0,9 \text{ MB}}$