

INDUSTRIA 4.0

Processamento de
Big Data



AVENTUREIROS DO BAIRRO PROIBIDO, FONTE: GOOGLE

Aula #9 - Transações em escala (BigData??)

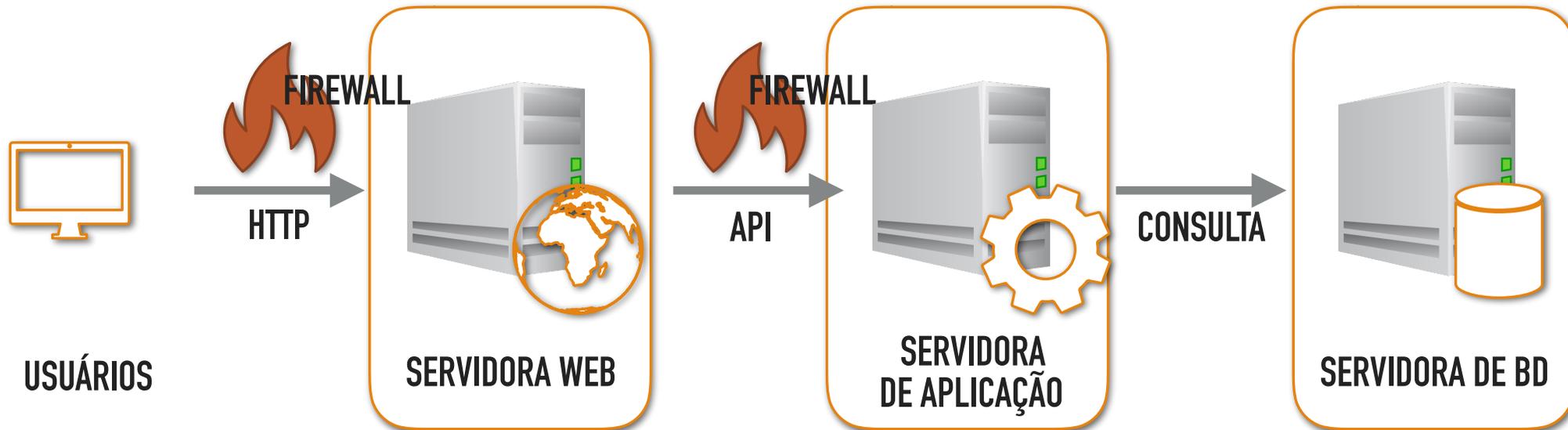
EDUARDO CUNHA DE ALMEIDA

Agenda

- Ideia 1: Comprar maquinas
- Ideia 2: Replicar o banco de dados
- Ideia 3: Cache de resultados
- Ideia 4: Stored Procedures
- Ideia 5: Particionar o banco de dados
- Ideia 6: Desistir do ACID

FONTE: FALOUTSOS E PAVLO, 2014

Ideia 1: Comprar máquina rápida

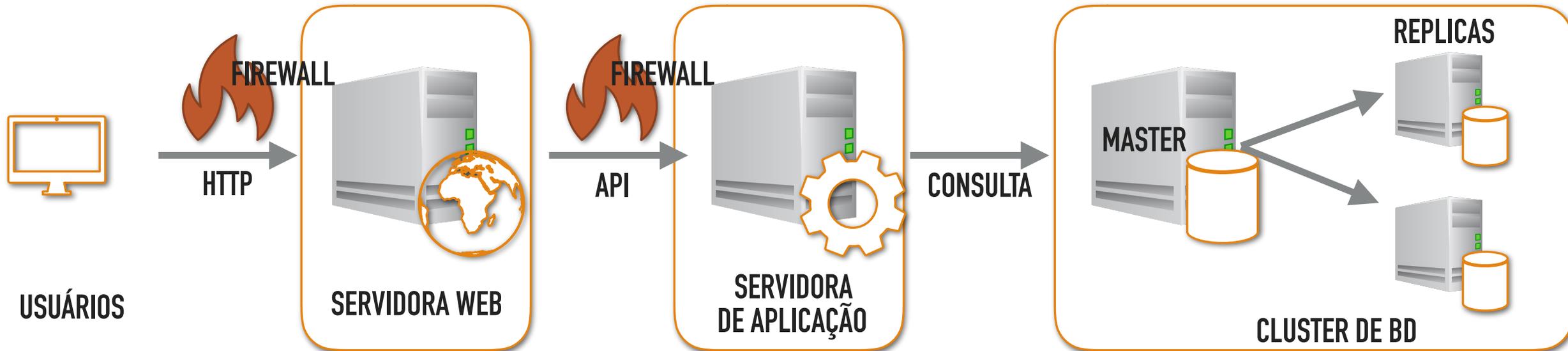


- HDD rápido
- DRAM
- CPUs
- SSD

(+) NÃO MUDA APLICAÇÃO
(+) MELHORIA IMEDIATA

(-) CARO!!!
(-) ÚNICO PONTO DE FALHA

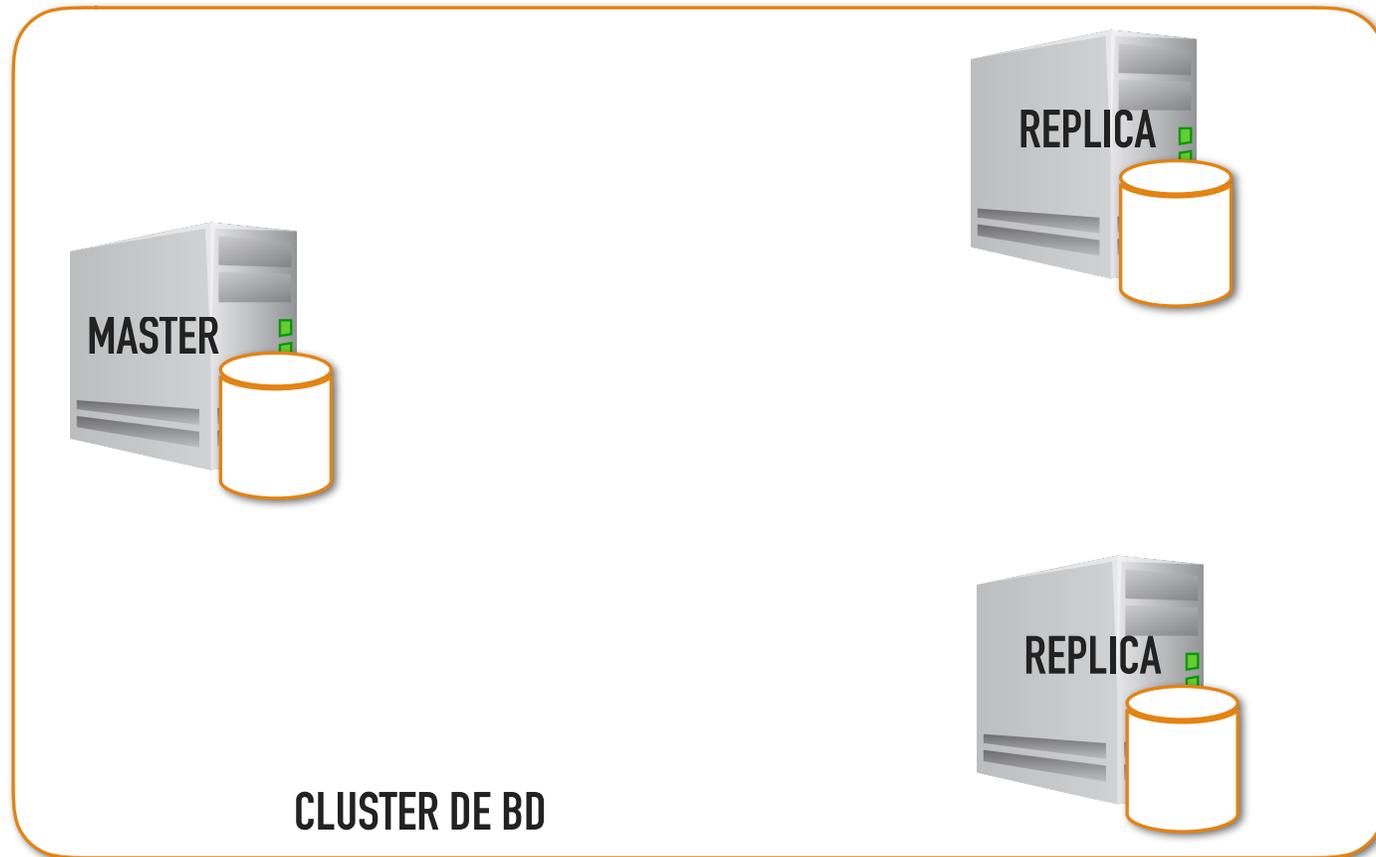
Ideia 2: Replicar o Banco de Dados



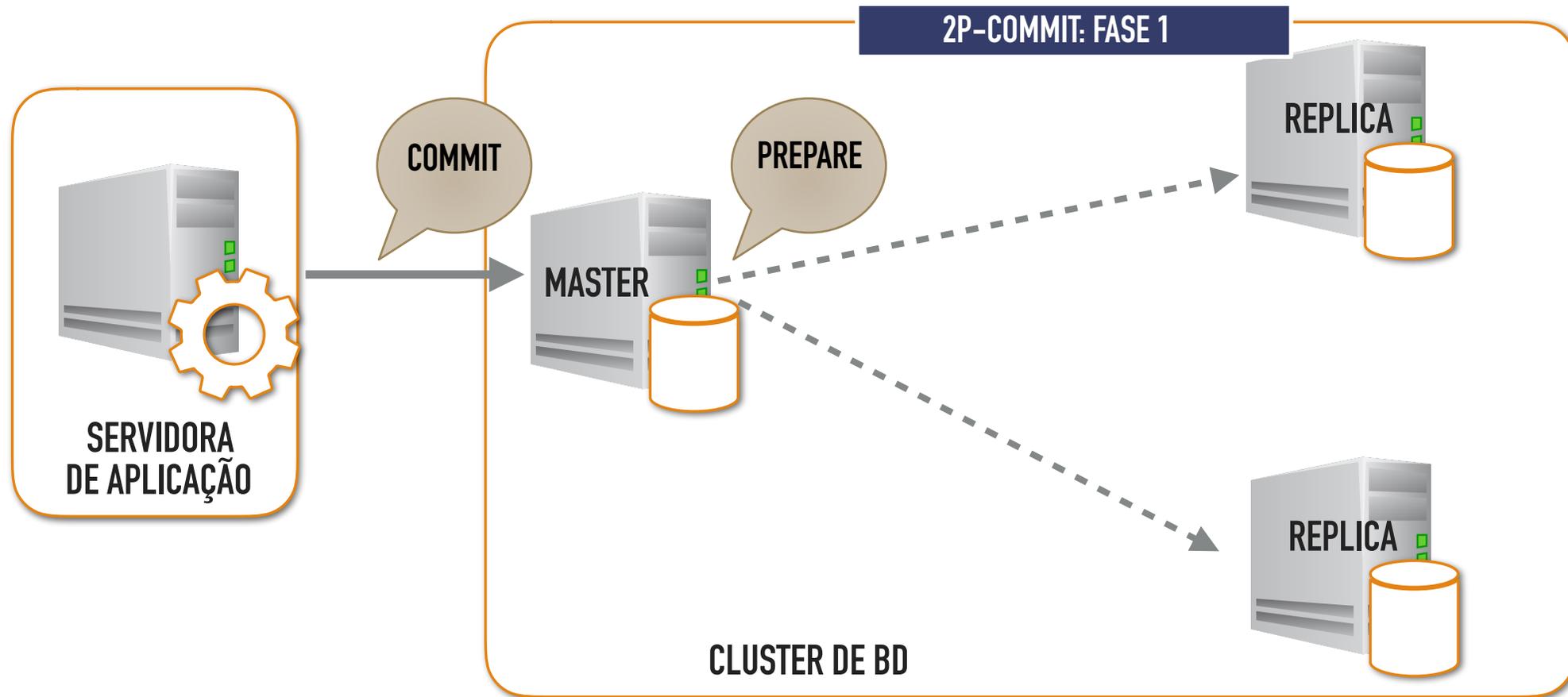
- (+) NÃO MUDA APLICAÇÃO
- (+) PARALELIZAÇÃO DE LEITURAS
- (+) MELHORA TOLERANCIA A FALHAS

- (-) CARO PRA MANTER REPLICAS E HW!!!
- (-) LATÊNCIA DE ESCRITAS = NODO MAIS LENTO

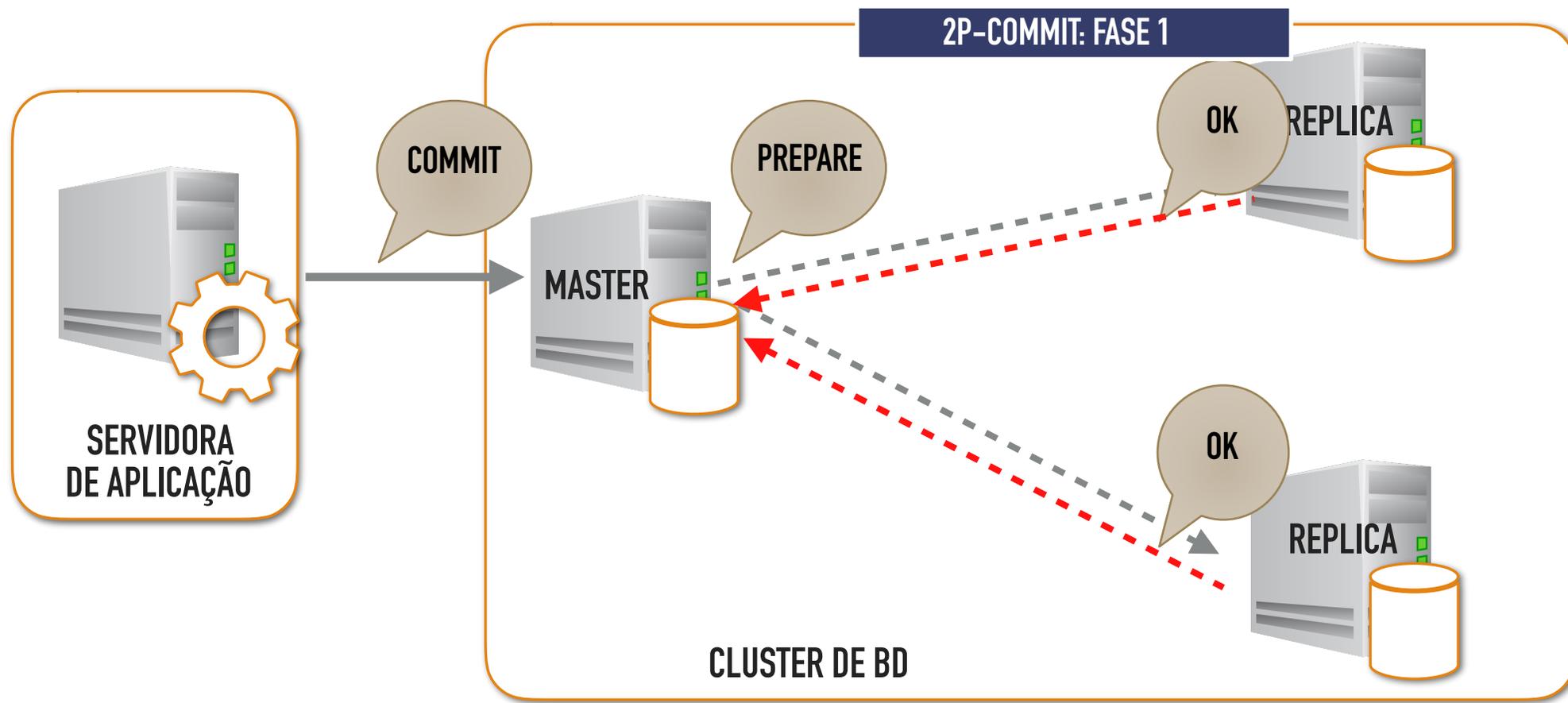
Ideia 2: Replicar o Banco de Dados



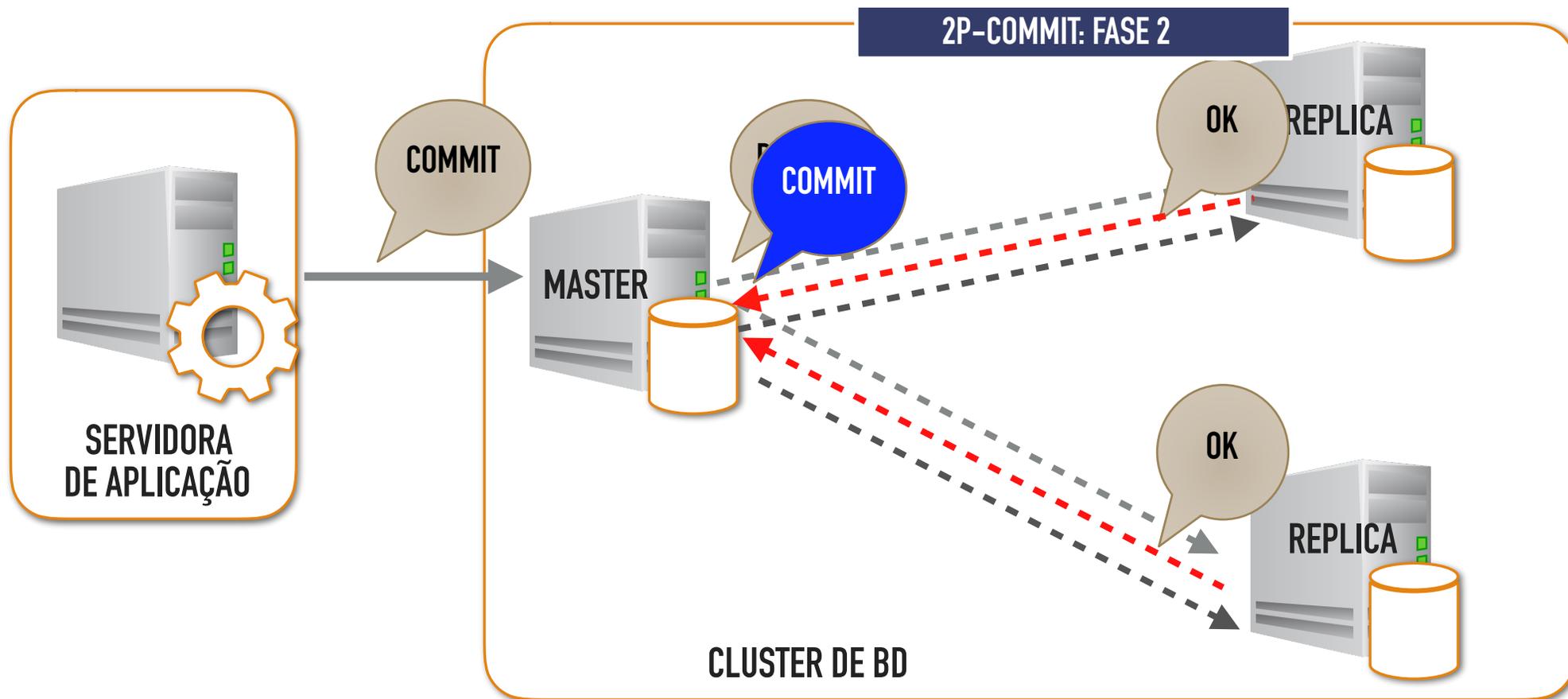
Ideia 2: Replicar o Banco de Dados



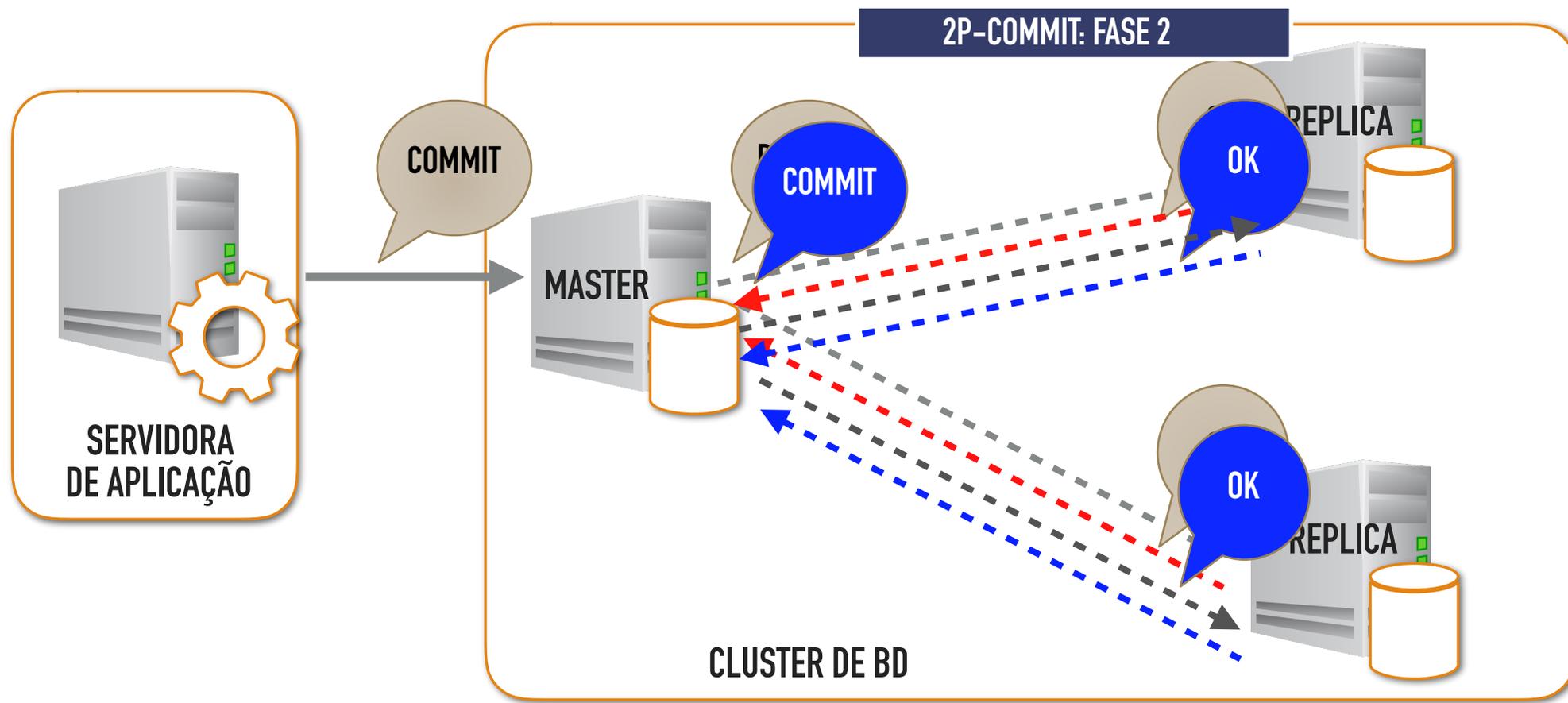
Ideia 2: Replicar o Banco de Dados



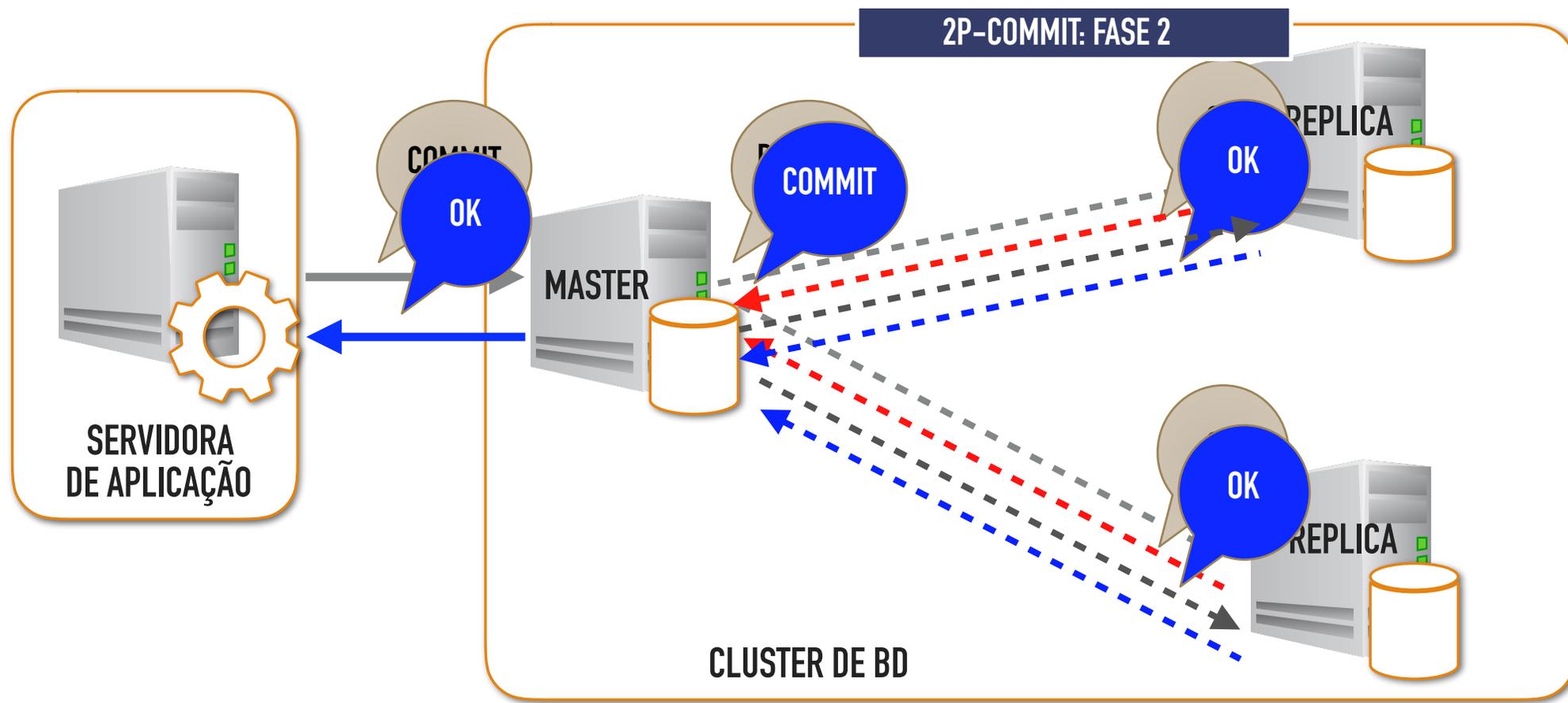
Ideia 2: Replicar o Banco de Dados



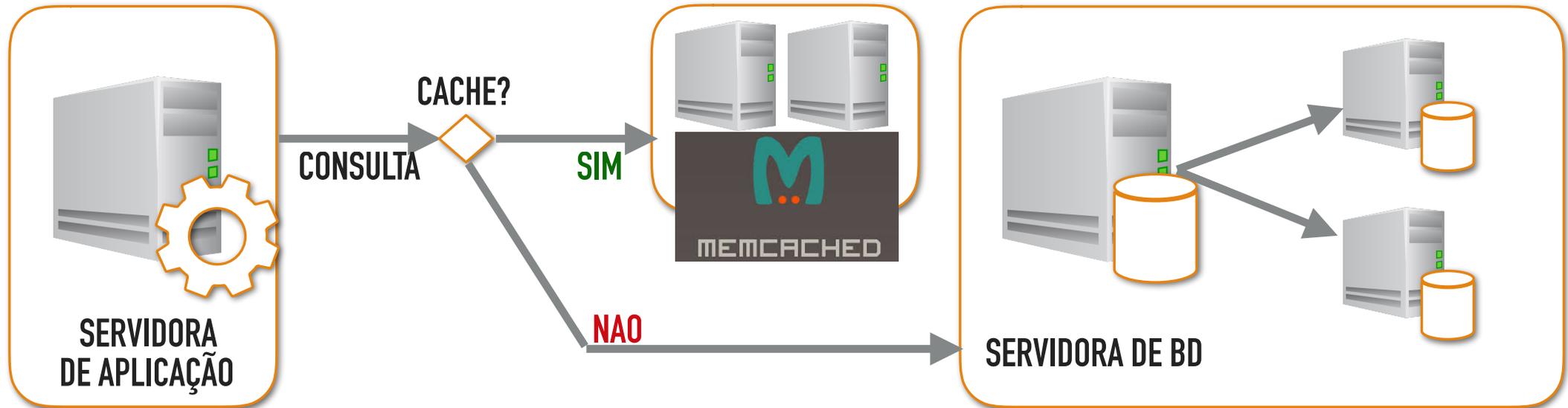
Ideia 2: Replicar o Banco de Dados



Ideia 2: Replicar o Banco de Dados



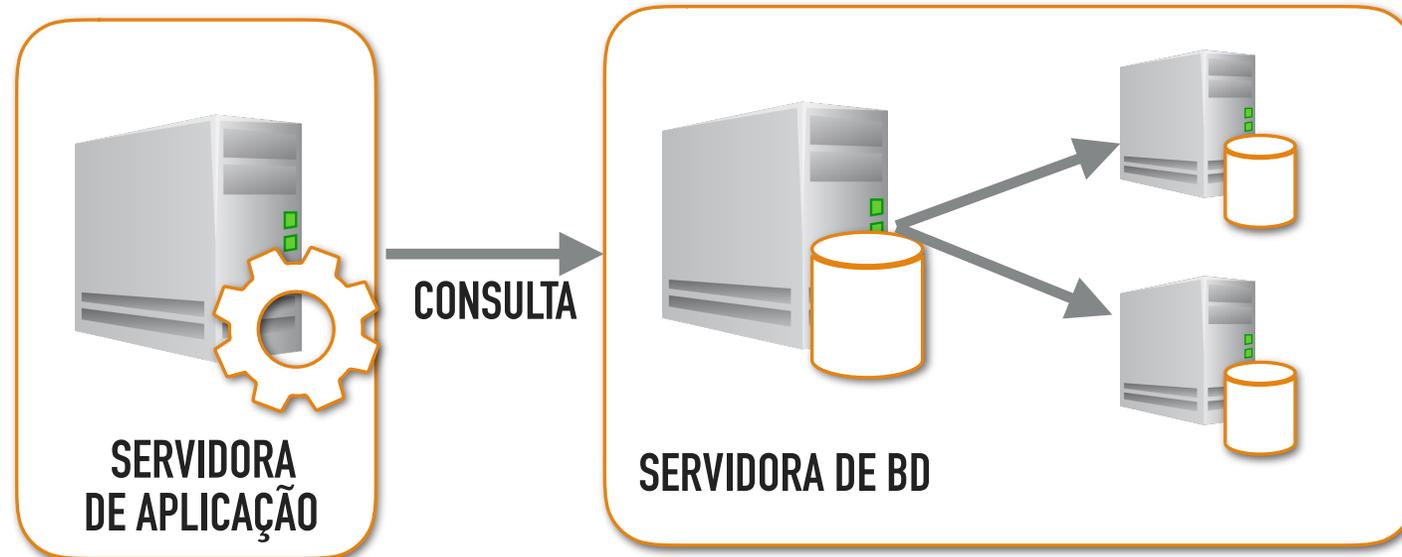
Ideia 3: Cache de resultados



(+) REDUZ CARGA NO BD
(+) API RÁPIDA

(-) ROTEAMENTO EXTRA DE MENSAGENS
(-) APLICAÇÃO DEVE SER ALTERADA
(-) PROBLEMAS COM VAZÃO DE ESCRITAS

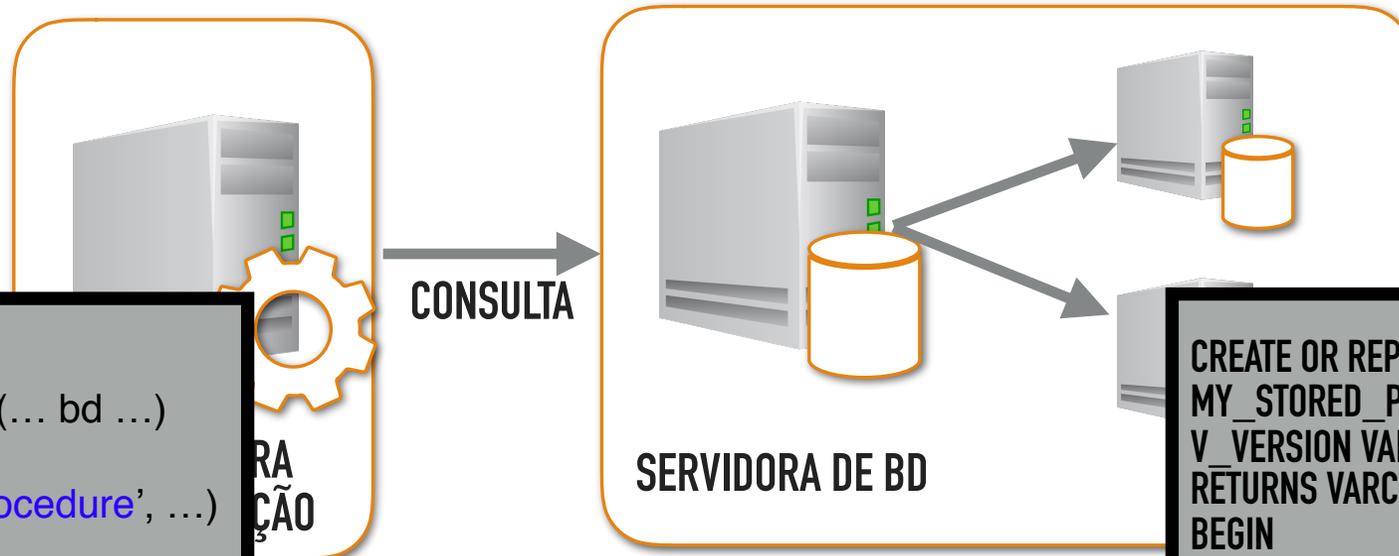
Ideia 4: Stored procedures



(+) REDUZ MENSAGENS DE REDE
(+) REDUZ CONTENÇÃO
(+) MODULARIZAÇÃO

(-) LÓGICA DA APLICAÇÃO EM VÁRIOS LOCAIS
(-) API NÃO PADRÃO
(-) APRISIONAMENTO TECNOLÓGICO (LOCK-IN)

Ideia 4: Stored procedures



try:

```
conn = MySQLdb.connect (... bd ...)  
c1=conn.cursor()  
c1.callproc('My_Stored_Procedure', ...)  
c1.close()
```

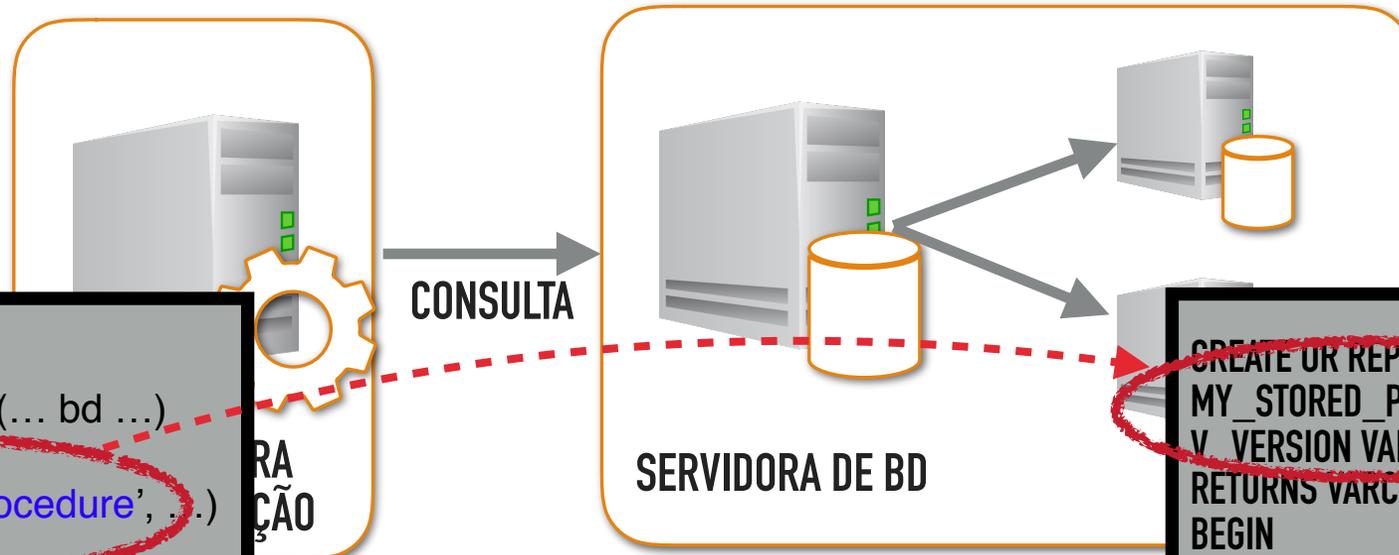
except (Exception) as error:
 print(error)

finally:

```
if conn is not None:  
    conn.close()
```

```
CREATE OR REPLACE FUNCTION  
MY_STORED_PROCEDURE(V_NAME VARCHAR,  
V_VERSION VARCHAR)  
RETURNS VARCHAR AS $$  
BEGIN  
    IF V_VERSION IS NULL THEN  
        RETURN V_NAME;  
    END IF;  
    RETURN V_NAME || ' ' || V_VERSION;  
END;  
$$ LANGUAGE PLPGSQL;
```

Ideia 4: Stored procedures



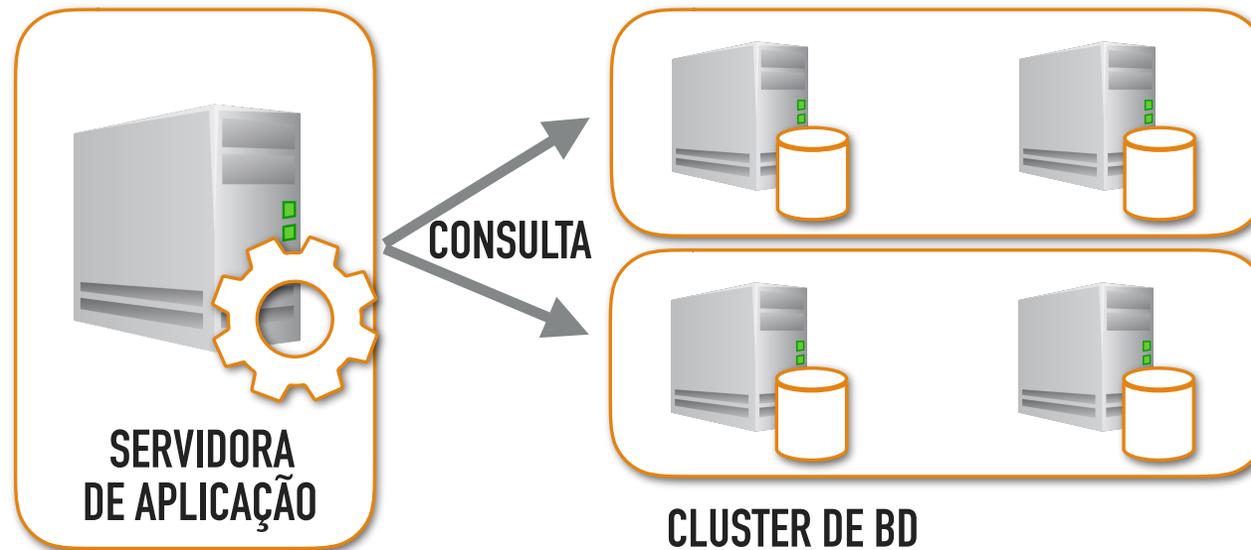
```
try:  
    conn = MySQLdb.connect (... bd ...)  
    c1=conn.cursor()  
    c1.callproc('My_Stored_Procedure', ...)  
    c1.close()
```

```
except (Exception) as error:  
    print(error)
```

```
finally:  
    if conn is not None:  
        conn.close()
```

```
CREATE OR REPLACE FUNCTION  
MY_STORED_PROCEDURE(V_NAME VARCHAR,  
V_VERSION VARCHAR)  
RETURNS VARCHAR AS $$  
BEGIN  
    IF V_VERSION IS NULL THEN  
        RETURN V_NAME;  
    END IF;  
    RETURN V_NAME || ' ' || V_VERSION;  
END;  
$$ LANGUAGE PLPGSQL;
```

Ideia 5: Particionar o banco de dados

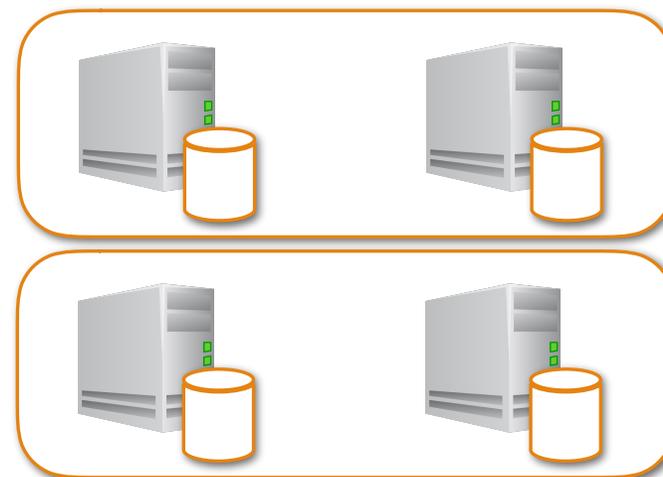


(+) PARALELIZAR TODAS AS OPERAÇÕES
(+) FÁCIL ADICIONAR/REMOVER HW

(-) MAIORIA DOS BDS NÃO FAZEM
(-) JUNÇÕES SÃO CARAS
(-) PARTICIONAMENTO NÃO É TRIVIAL

<u>cliente. nome_c</u>	<u>cesta. dt_compra</u>	<u>compra. isbn</u>
André	10/10/17	1234
Cido	11/10/17	1121
Siloé	20/10/17	1234
Siloé	20/10/17	2222
Siloé	20/10/17	2223
Juliana	20/10/17	1234
Juliana	20/10/17	2224
Rosana	22/10/17	2225
Rosana	22/10/17	1234
Rosana	22/10/17	1121
Rosana	22/10/17	2222

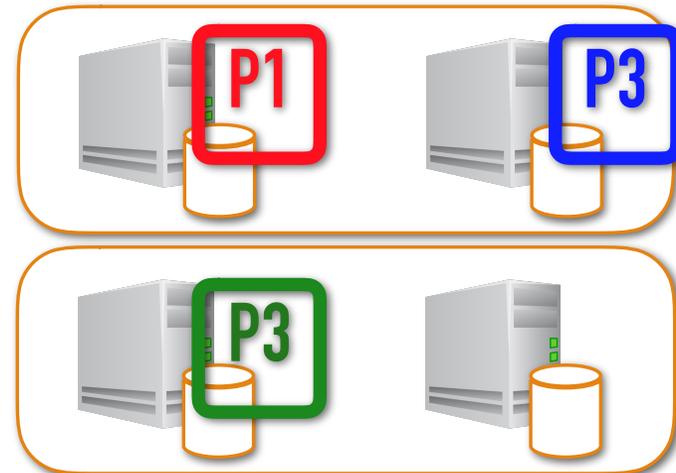
Integrar o banco de dados



CLUSTER DE BD

Particionar o banco de dados

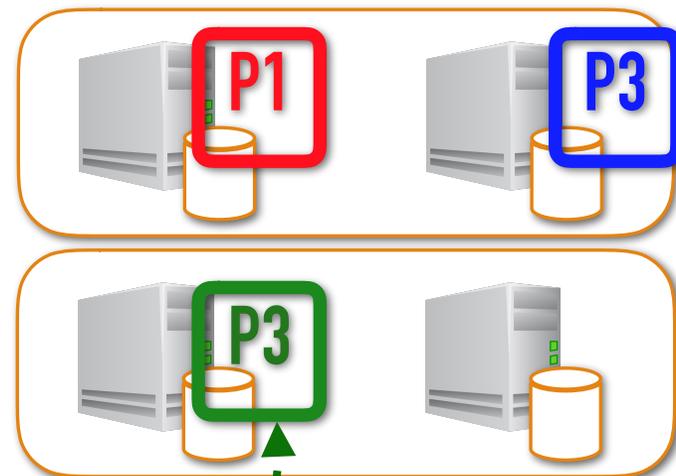
	<u>cliente. nome_c</u>	<u>cesta. dt_compra</u>	<u>compra. isbn</u>
P1	Andre	11/10/17	1234
	Cido	11/10/17	1121
P2	Siloé	20/10/17	1234
	Siloé	20/10/17	2222
	Siloé	20/10/17	2223
	Juliana	20/10/17	1234
	Juliana	20/10/17	2224
P3	Rosana	22/10/17	2225
	Rosana	22/10/17	1234
	Rosana	22/10/17	1121
	Rosana	22/10/17	2222



CLUSTER DE BD

<u>cliente. nome_c</u>	<u>cesta. dt_compra</u>	<u>compra. isbn</u>
André	11/10/17	1234
Cido	11/10/17	1121
Siloé	20/10/17	1234
Siloé	20/10/17	2222
Siloé	20/10/17	2223
Juliana	20/10/17	1234
Juliana	20/10/17	2224
Rosana	22/10/17	2225
Rosana	22/10/17	1234
Rosana	22/10/17	1121
Rosana	22/10/17	2222
Rosana	22/10/17	2223
Rosana	22/10/17	2224

Atualizar o banco de dados

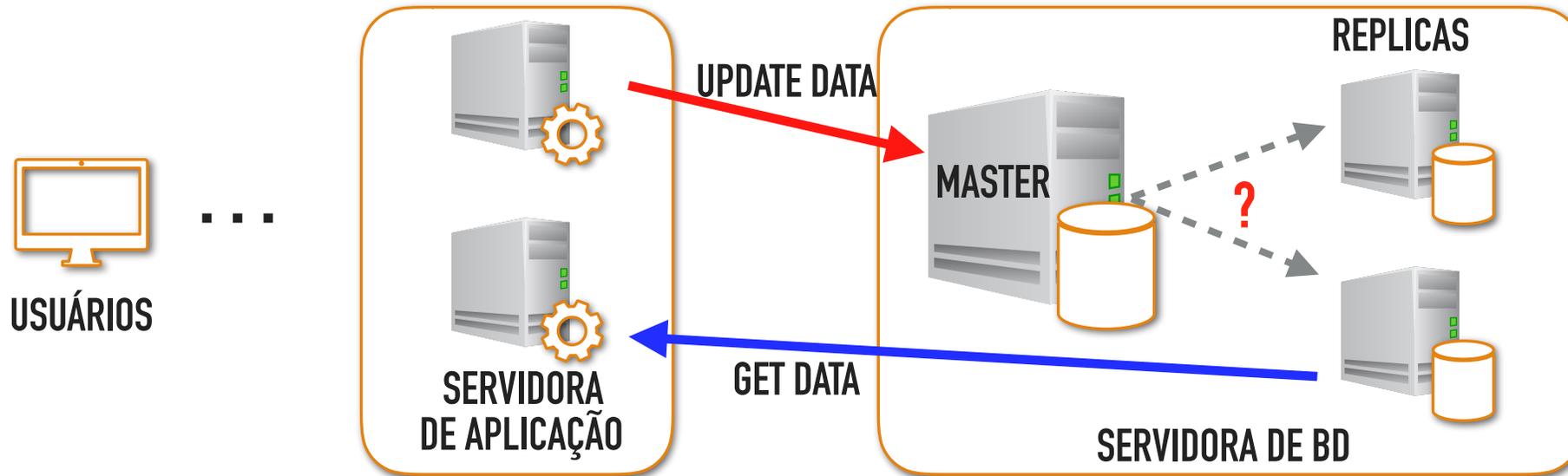


CLUSTER DE BD

```
BEGIN
GET CLIENTE.NOME_C="ROSANA"
// CODIGO INCLUSÃO...
COMMIT
```

P3

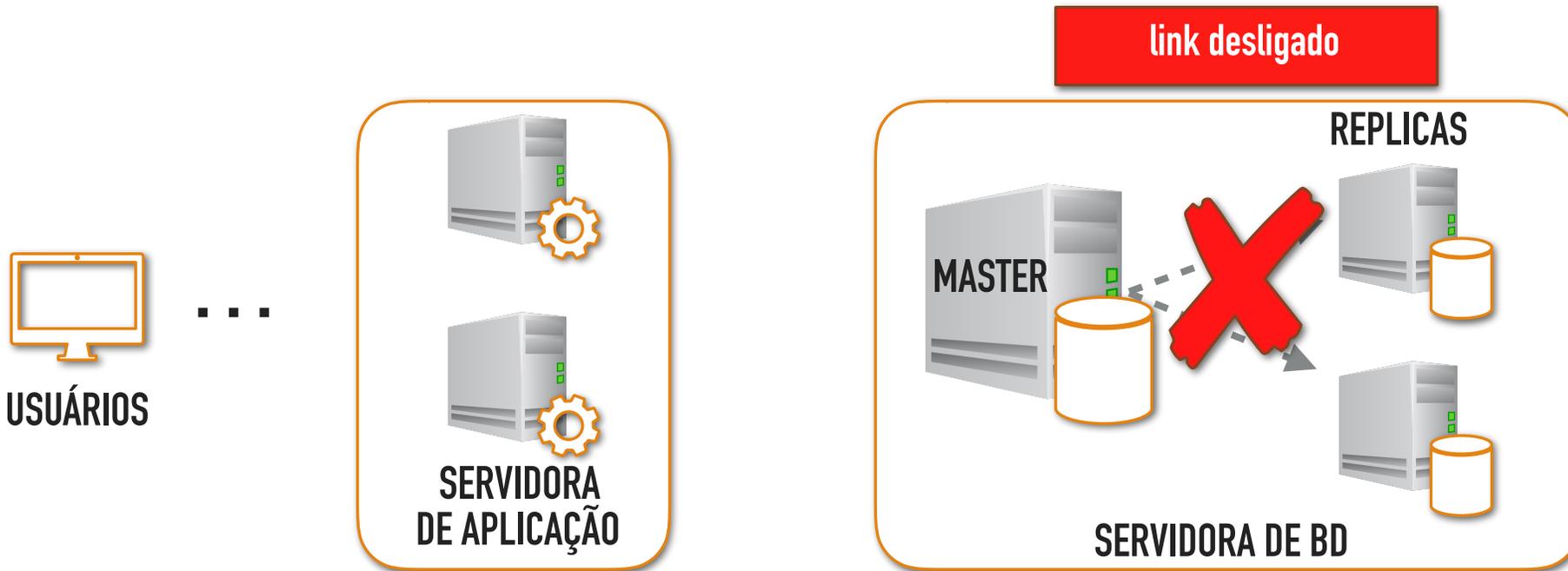
Ideia 6: Desistir do ACID



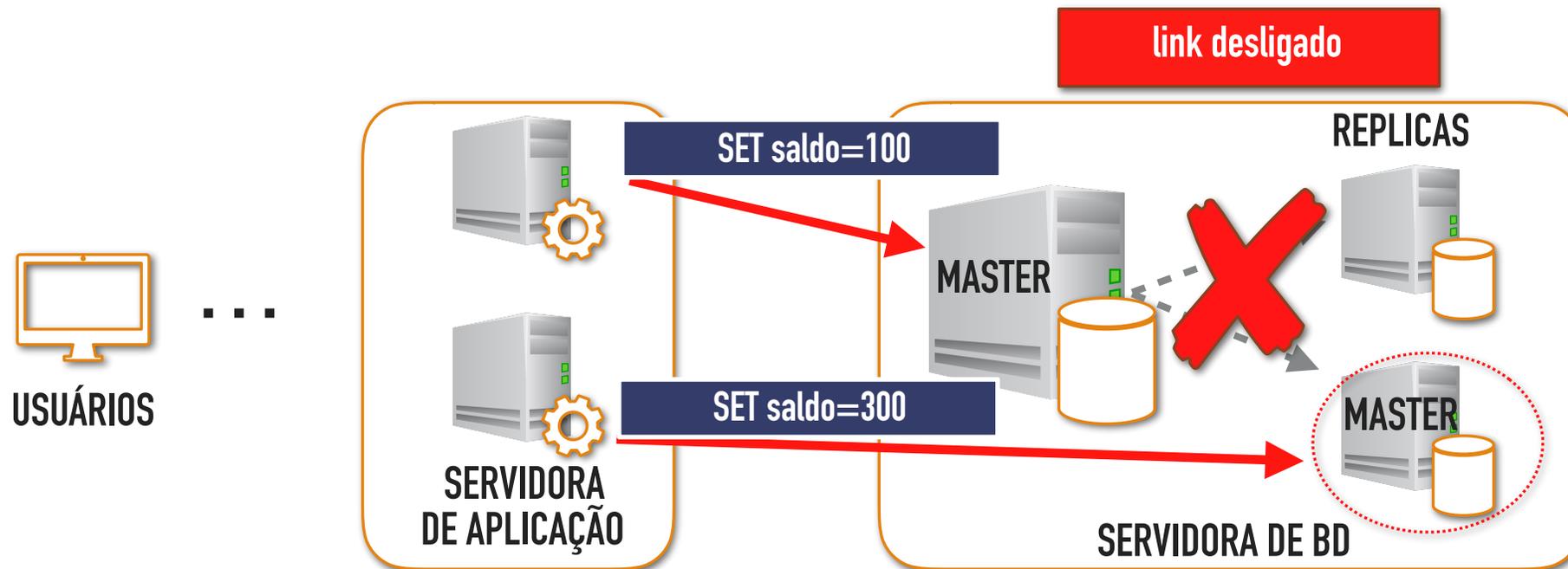
(+) ESCALABILIDADE
(+) PARALELISMO

(-) PÉSSIMA IDEIA
(-) POSSÍVEL PERDA DE ATUALIZAÇÕES
(-) DISCIPLINA DOS PROGRAMADORES

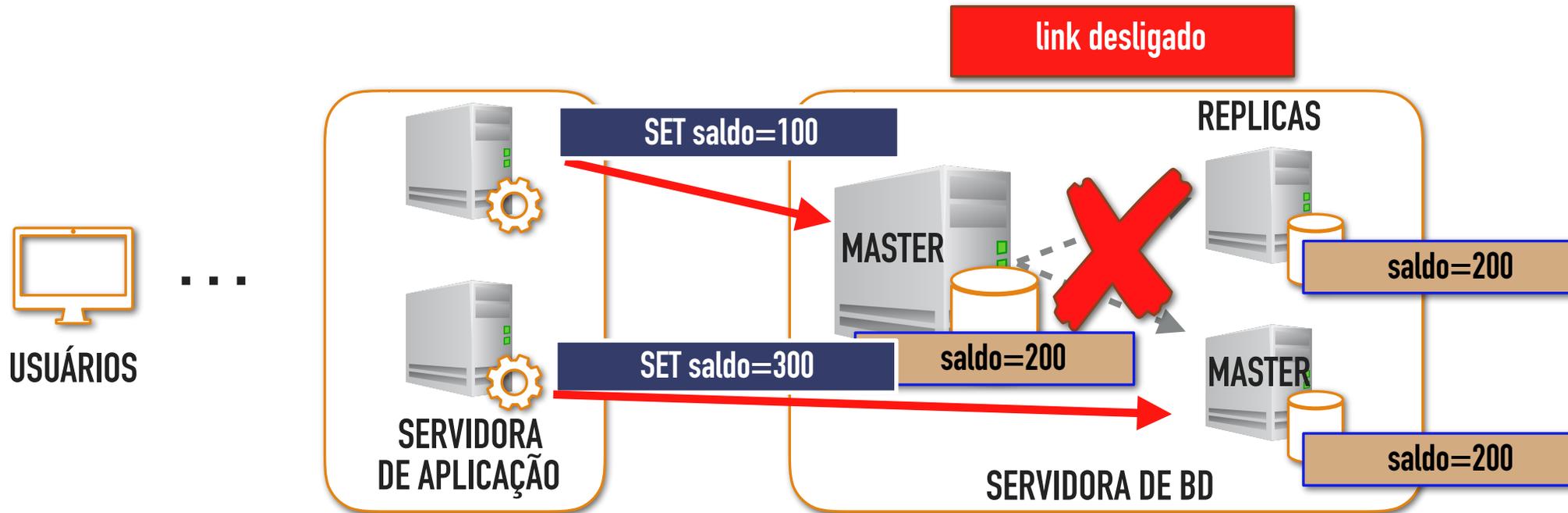
Ideia 6: Desistir do ACID



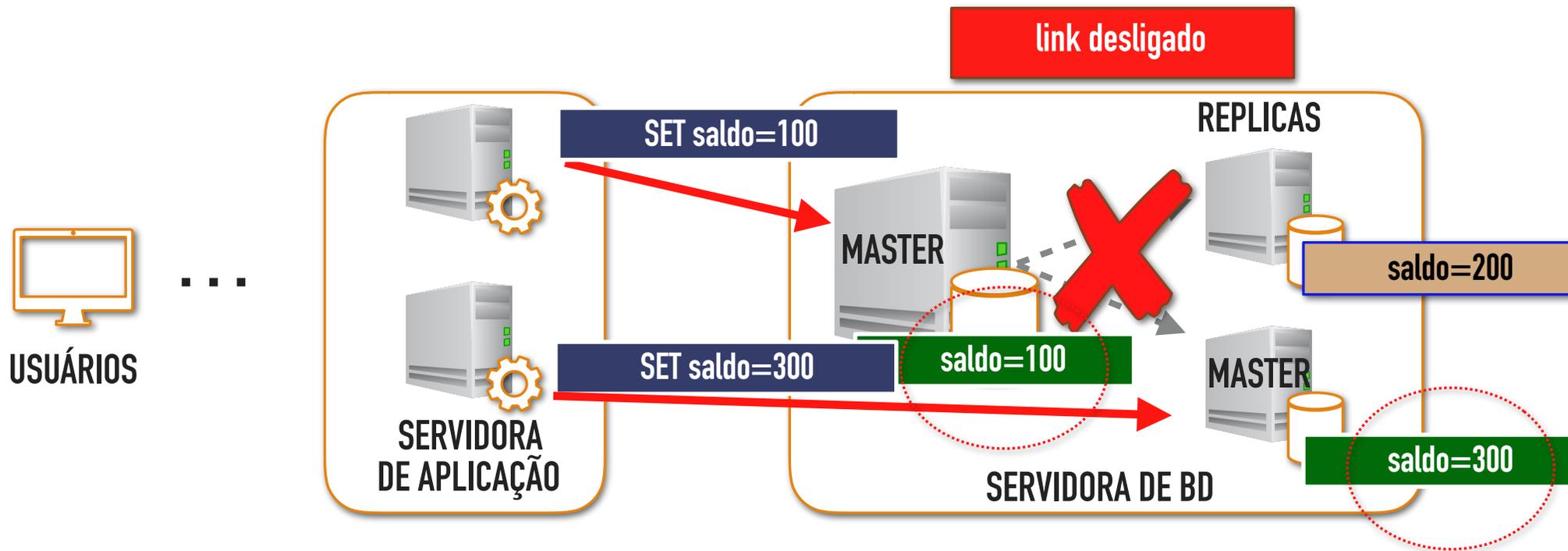
Ideia 6: Desistir do ACID



Ideia 6: Desistir do ACID



Ideia 6: Desistir do ACID



INDUSTRIA 4.0

Processamento de
Big Data



AVENTUREIROS DO BAIRRO PROIBIDO, FONTE: GOOGLE

Aula #9 - Transações em escala (BigData??)

EDUARDO CUNHA DE ALMEIDA