

# NSR Calculation Requirements – Mina Caraíba

## 1. Visão Geral

Este documento especifica os requisitos de cálculo do NSR (Net Smelter Return) baseado na planilha "**NSR Caraíba - V.1\_open.xlsxm**". O sistema deve replicar fielmente os cálculos dessa planilha.

## 2. Estrutura de Dados

### 2.1 Metais Suportados

O sistema deve suportar cálculo multi-metal com:

Metal	Código	Unidade de Teor	Unidade de Preço
Cobre	Cu	%	\$/lb
Ouro	Au	g/t	\$/oz
Prata	Ag	g/t	\$/oz

**Nota:** Au e Ag são tratados como **subprodutos** do concentrado de cobre (notação "Au -> Cu", "Ag -> Cu").

### 2.2 Áreas de Mineração (Minas)

O sistema deve suportar múltiplas minas/áreas:

Mina	Áreas
Pilar UG	Deepening Above-965, Deepening Below-965, MSBSUL, P1P2NE, P1P2W, BARAUNA, HONEYPO, R22UG, MSBW, GO2040, PROJETO N-100, EAST LIMB

Mina	Áreas
Vermelhos UG	Vermelhos Sul, UG03, N5/UG04, N8-UG
Surubim & C12	Surubim OP, C12 OP, C12 UG
Vermelhos OP	N8, N9
Suçuarana OP	Suçuarana OP, S10, S5

## 2.3 Métodos de Mineração

Método	Código
Underground (Subterrâneo)	UG
Open Pit (Céu Aberto)	OP

## 3. Inputs do Sistema

### 3.1 Parâmetros de Minério (Ore)

Campo	Descrição	Unidade	Exemplo
ore_tonnage	Massa de minério	tonnes	20,000
mine_dilution	Diluição da mina	decimal (0-1)	0.14
ore_recovery	Recuperação de minério	decimal (0-1)	0.98

### 3.2 Teores de Cabeça (Head Grades)

Campo	Descrição	Unidade	Exemplo
cu_grade	Teor de cobre	%	1.4

Campo	Descrição	Unidade	Exemplo
au_grade	Teor de ouro	g/t	0.23
ag_grade	Teor de prata	g/t	2.33

### 3.3 Price Deck (Preços de Commodities)

Campo	Descrição	Unidade	Exemplo (Mineral Resources)
cu_price	Preço do cobre	\$/lb	9,149 (centavos) = \$4.15/lb
au_price	Preço do ouro	\$/oz	2,400
ag_price	Preço da prata	\$/oz	29

#### Cenários de preço disponíveis:

- Mineral Resources
- Mineral Reserves
- Consensus Low
- Consensus Mean
- Consensus High

### 3.4 Termos Comerciais (Pricing Assumptions)

#### 3.4.1 Cobre (Cu)

Campo	Descrição	Valor Default	Unidade
cu_discount	Desconto no preço	3.35%	decimal
cu_payability	Payability	96.65%	decimal
cu_tc	Treatment Charge	40	\$/dmt conc
cu_rc	Refining Charge	1.90	\$/lb Cu pagável
cu_freight	Frete	84	\$/dmt conc
cu_penalties	Penalidades	0	\$/dmt conc
cu_other_costs	Outros custos	0	\$/dmt conc

Campo	Descrição	Valor Default	Unidade
conc_loss_factor	Fator de perda de concentrado	0	decimal

### 3.4.2 Ouro (Au -> Cu)

Campo	Descrição	Valor Default	Unidade
au_payability	Payability	90%	decimal
au_rc	Refining Charge	4.00	\$/oz Au pagável

### 3.4.3 Prata (Ag -> Cu)

Campo	Descrição	Valor Default	Unidade
ag_payability	Payability	90%	decimal
ag_rc	Refining Charge	0.35	\$/oz Ag pagável

## 3.5 Recuperação Metalúrgica

A recuperação de cobre é calculada por uma **fórmula linear** dependente do teor:

$$\text{Cu Recovery (\%)} = a \times \text{Cu Grade (\%)} + b$$

Parâmetros por área:

Área	a	b	Fixed (se aplicável)
Vermelhos Sul	2.8286	92.584	-
UG03	2.8286	92.584	-
Pilar UG - Deepening Above-965	4.0851	90.346	92.9%
MSBSUL	7.5986	85.494	90%
P1P2NE	2.3826	91.442	-
P1P2W	8.8922	87.637	-
R22UG	3.0368	91.539	-

Área	a	b	Fixed (se aplicável)
MSBW	3.0368	91.539	-
GO2040	5.4967	88.751	-
EAST LIMB	-	91	fixo
Surubim OP	4.0718	87.885	-
C12 OP	4.0718	87.885	-

**Nota:** Se "Fixed" está definido, usar o valor fixo em vez da fórmula.

Recuperação de Au e Ag: Configurável por cenário (ex: "Base Case" = valor fixo).

## 3.6 Teor do Concentrado

Campo	Descrição	Unidade	Exemplo
cu_conc_grade	Teor de Cu no concentrado	%	Configurável por cenário

## 3.7 Royalties

Tipo	Taxa	Descrição
CFEM	2%	Royalty governamental brasileiro
Third-Party	0%	Royalty para terceiros

## 3.8 Custos Operacionais (para EBITDA)

Campo	Descrição	Unidade	Exemplo
mine_cost	Custo de lavra	\$/t ore	28
development_cost	Custo de desenvolvimento	\$/meter	2,257
development_meters	Metros de desenvolvimento	meters	50
haul_cost	Custo de transporte	\$/t ore	13.57
plant_cost	Custo de planta	\$/t ore	7.4

Campo	Descrição	Unidade	Exemplo
ga_cost	G&A	\$/t ore	configurável

## 4. Fórmulas de Cálculo

### 4.1 Conversões de Unidades

```
# Constantes
TROY_OZ_PER_GRAM = 1 / 31.1035
LB_PER_TONNE = 2204.62
KG_PER_TONNE = 1000

# Conversão de teor para fração
def grade_to_fraction(grade: float, unit: str) -> float:
    if unit == "%":
        return grade / 100
    elif unit == "g/t":
        return grade / 1_000_000 # g/t = ppm
    elif unit == "ppm":
        return grade / 1_000_000
    return grade

# Conversão de preço
def price_per_lb_to_per_tonne(price_per_lb: float) -> float:
    return price_per_lb * LB_PER_TONNE

def price_per_oz_to_per_gram(price_per_oz: float) -> float:
    return price_per_oz * TROY_OZ_PER_GRAM
```

## 4.2 Cálculo de Recuperação de Cobre

```
def compute_cu_recovery(cu_grade_pct: float, area: str) -> float:  
    """  
    Calcula recuperação de cobre baseado no teor e área.  
    Retorna valor em decimal (0-1).  
    """  
  
    params = get_recovery_params(area)  
  
    if params.fixed is not None:  
        return params.fixed / 100  
  
    recovery_pct = params.a * cu_grade_pct + params.b  
    return min(recovery_pct / 100, 1.0) # Cap at 100%
```

## 4.3 Cálculo de Metal Contido no Concentrado

```
def compute_metal_in_concentrate(
    ore_tonnage: float,
    head_grade: float, # em unidade original (%, g/t)
    grade_unit: str,
    recovery: float, # decimal
    cu_conc_grade: float, # %
) -> dict:
    """
    Calcula a quantidade de metal contido no concentrado.
    """

    # Metal contido no minério
    if grade_unit == "%":
        metal_in_ore = ore_tonnage * (head_grade / 100)
    elif grade_unit == "g/t":
        metal_in_ore = ore_tonnage * head_grade / 1000 # kg

    # Metal recuperado
    metal_recovered = metal_in_ore * recovery

    # Tonelagem de concentrado (para Cu)
    if grade_unit == "%": # Cobre
        conc_tonnage = metal_recovered / (cu_conc_grade / 100)
    else:
        conc_tonnage = None # Au/Ag não determinam tonnage de conc

    return {
        "metal_in_ore": metal_in_ore,
        "metal_recovered": metal_recovered,
        "conc_tonnage": conc_tonnage
    }
```

## 4.4 Cálculo de Preço do Concentrado (Conc. Price)

```
def compute_conc_price_cu(
    cu_price_per_lb: float,
    cu_conc_grade: float, # %
    payability: float, # decimal (0.9665)
    tc: float, # $/dmt
    rc: float, # $/lb
    freight: float, # $/dmt
    penalties: float = 0, # $/dmt
    other_costs: float = 0, # $/dmt
) -> float:
    """
    Calcula o preço do concentrado de Cu em $/t conc.

    Fórmula:
    Conc Price = (Cu Price × Cu Conc Grade × Payability × LB_PER_TONNE) – TC – (RC × Cu
    """
    cu_grade_fraction = cu_conc_grade / 100

    # Receita bruta por tonelada de concentrado
    gross_revenue = cu_price_per_lb * cu_grade_fraction * payability * LB_PER_TONNE

    # Deduções
    rc_total = rc * cu_grade_fraction * LB_PER_TONNE # RC por lb de Cu no conc
    total_deductions = tc + rc_total + freight + penalties + other_costs

    return gross_revenue - total_deductions

def compute_conc_price_au(
    au_price_per_oz: float,
    au_grade_in_conc: float, # g/t no concentrado
    payability: float, # decimal (0.90)
    rc: float, # $/oz
) -> float:
    """
    Calcula a contribuição do Au para o preço do concentrado em $/t conc.

    """
    au_oz_per_tonne_conc = au_grade_in_conc * TROY_OZ_PER_GRAM

    gross_revenue = au_price_per_oz * au_oz_per_tonne_conc * payability
    rc_total = rc * au_oz_per_tonne_conc
```

```
return gross_revenue - rc_total

def compute_conc_price_ag(
    ag_price_per_oz: float,
    ag_grade_in_conc: float, # g/t no concentrado
    payability: float, # decimal (0.90)
    rc: float, # $/oz
) -> float:
    """
    Calcula a contribuição do Ag para o preço do concentrado em $/t conc.
    """
    ag_oz_per_tonne_conc = ag_grade_in_conc * TROY_OZ_PER_GRAM

    gross_revenue = ag_price_per_oz * ag_oz_per_tonne_conc * payability
    rc_total = rc * ag_oz_per_tonne_conc

    return gross_revenue - rc_total
```

## 4.5 Cálculo do NSR por Tonelada de Minério

```
def compute_nsr_per_tonne(
    conc_price_total: float, # $/t conc (Cu + Au + Ag)
    conc_ratio: float, # t conc / t ore
) -> float:
    """
    Calcula NSR por tonelada de minério.
    
```

$$\text{NSR } (\$/\text{t ore}) = \text{Conc Price } (\$/\text{t conc}) \times \text{Conc Ratio } (\text{t conc} / \text{t ore})$$

Onde:

$$\text{Conc Ratio} = (\text{Cu Grade} \times \text{Cu Recovery}) / \text{Cu Conc Grade}$$

.....

```
return conc_price_total * conc_ratio
```

```
def compute_conc_ratio(
    cu_grade: float, # %
    cu_recovery: float, # decimal
    cu_conc_grade: float, # %
) -> float:
    """
    Calcula a razão de concentrado (t conc / t ore).
    """

    return (cu_grade / 100) * cu_recovery / (cu_conc_grade / 100)
```

## 4.6 Ajustes de Diluição e Recuperação de Minério

```
def compute_nsr_with_mine_factors(  
    nsr_processing: float, # NSR após processing ($/t ore)  
    mine_dilution: float, # decimal (0.14)  
    ore_recovery: float, # decimal (0.98)  
) -> float:  
    """  
        Ajusta NSR para fatores de mina (diluição e recuperação).  
    """
```

NSR Mine = NSR Processing × (1 – Dilution) × Ore Recovery

Alternativamente, pode ser calculado como:

```
NSR Mine = Mineral Resources NSR – Dilution & Ore Loss  
"""
```

```
dilution_factor = (1 – mine_dilution) * ore_recovery  
return nsr_processing * dilution_factor
```

## 4.7 Cálculo Completo do NSR

```
def compute_nsr_complete(inputs: NSRInputs) -> NSRResult:  
    """  
    Cálculo completo do NSR seguindo a metodologia Caraíba.  
    """  
  
    # 1. Calcular recuperação de Cu  
    cu_recovery = compute_cu_recovery(inputs.cu_grade, inputs.area)  
  
    # 2. Calcular razão de concentrado  
    conc_ratio = compute_conc_ratio(  
        inputs.cu_grade,  
        cu_recovery,  
        inputs.cu_conc_grade  
    )  
  
    # 3. Calcular teores no concentrado (Au e Ag)  
    # Au e Ag no conc = (teor no ore × recovery) / conc_ratio  
    au_in_conc = (inputs.au_grade * inputs.au_recovery) / conc_ratio  
    ag_in_conc = (inputs.ag_grade * inputs.ag_recovery) / conc_ratio  
  
    # 4. Calcular preço do concentrado  
    conc_price_cu = compute_conc_price_cu(  
        inputs.cu_price, inputs.cu_conc_grade,  
        inputs.cu_payability, inputs.cu_tc, inputs.cu_rc,  
        inputs.cu_freight, inputs.cu_penalties  
    )  
  
    conc_price_au = compute_conc_price_au(  
        inputs.au_price, au_in_conc,  
        inputs.au_payability, inputs.au_rc  
    )  
  
    conc_price_ag = compute_conc_price_ag(  
        inputs.ag_price, ag_in_conc,  
        inputs.ag_payability, inputs.ag_rc  
    )  
  
    conc_price_total = conc_price_cu + conc_price_au + conc_price_ag  
  
    # 5. Calcular NSR por tonelada de minério (nível Mineral Resources)  
    nsr_cu = conc_price_cu * conc_ratio  
    nsr_au = conc_price_au * conc_ratio
```

```

nsr_ag = conc_price_ag * conc_ratio
nsr_total = nsr_cu + nsr_au + nsr_ag

# 6. Aplicar fatores de mina (diluição e ore recovery)
nsr_mine = compute_nsr_with_mine_factors(
    nsr_total,
    inputs.mine_dilution,
    inputs.ore_recovery
)

return NSRResult(
    # Preço do concentrado
    conc_price_cu=conc_price_cu,
    conc_price_au=conc_price_au,
    conc_price_ag=conc_price_ag,
    conc_price_total=conc_price_total,

    # NSR por componente ($/t ore)
    nsr_cu=nsr_cu,
    nsr_au=nsr_au,
    nsr_ag=nsr_ag,

    # NSR total
    nsr_mineral_resources=nsr_total, # Antes de diluição
    nsr_mine=nsr_mine, # Após diluição e ore recovery

    # Breakdown
    conc_ratio=conc_ratio,
    cu_recovery=cu_recovery,
    dilution_loss=nsr_total - nsr_mine,

    # Metadados
    inputs_used=inputs,
    formula_applied="NSR = Σ(Conc Price × Conc Ratio) × (1 - Dilution) × Ore Recovery"
)

```

## 5. Outputs do Sistema

### 5.1 Preço do Concentrado (\$/t conc)

Output	Descrição	Exemplo
conc_price_cu	Contribuição do Cu	\$2,824.68
conc_price_au	Contribuição do Au	\$244.76
conc_price_ag	Contribuição do Ag	\$29.65
conc_price_total	Preço total	\$3,099.09

### 5.2 NSR por Tonelada de Minério (\$/t ore)

Output	Descrição	Exemplo
nsr_cu	NSR do Cu	\$108.21
nsr_au	NSR do Au	\$9.38
nsr_ag	NSR do Ag	\$1.14
nsr_total	NSR Total (Mineral Resources)	\$118.72
nsr_processing	NSR após Processing	\$131.76
nsr_mine	NSR após Mina (com diluição)	\$148.01

### 5.3 Breakdown

Output	Descrição
dilution_ore_loss	Perda por diluição e ore recovery
recovery_loss	Perda por recuperação metalúrgica
freight_and_others	Custos de frete e outros

## 5.4 EBITDA (Opcional)

Output	Descrição	Fórmula
revenue	Receita total	NSR × Tonnage
mine_cost_total	Custo de mina total	Mine Cost × Tonnage
dev_cost_total	Custo de desenvolvimento	Dev Cost × Meters
haul_cost_total	Custo de transporte	Haul Cost × Tonnage
plant_cost_total	Custo de planta	Plant Cost × Tonnage
ebitda	EBITDA	Revenue - Custos

## 6. Validação

### 6.1 Caso de Teste: Vermelhos Sul

#### Inputs:

- Mine: Vermelhos UG
- Area: Vermelhos Sul
- Cu Grade: 1.4%
- Au Grade: 0.23 g/t
- Ag Grade: 2.33 g/t
- Mine Dilution: 14%
- Ore Recovery: 98%
- Cu Price: \$9,149/lb (ou valor equivalente)
- Au Price: \$2,400/oz
- Ag Price: \$29/oz
- Cu Payability: 96.65%
- Cu TC: \$40/dmt
- Cu RC: \$1.90/lb
- Cu Freight: \$84/dmt

#### Outputs Esperados:

- Conc Price Cu: ~\$2,824.68/t conc

- Conc Price Au: ~\$244.76/t conc
- Conc Price Ag: ~\$29.65/t conc
- Conc Price Total: ~\$3,099.09/t conc
- NSR Cu: ~\$108.21/t ore
- NSR Au: ~\$9.38/t ore
- NSR Ag: ~\$1.14/t ore
- NSR Total: ~\$118.72/t ore
- Mineral Resources: ~\$175.61
- NSR Mine: ~\$148.01

**Tolerância:** Diferença ≤ 0.1% (exceto arredondamento)

## 7. Requisitos Funcionais

### REQ-001: Suporte Multi-Metal

O sistema DEVE suportar cálculo de NSR para:

- Metal principal: Cobre (Cu)
- Subprodutos: Ouro (Au) e Prata (Ag)

### REQ-002: Termos Comerciais Configuráveis

O sistema DEVE permitir configuração de:

- Payability por metal
- TC (Treatment Charge) para Cu
- RC (Refining Charge) por metal
- Freight
- Penalidades

### REQ-003: Recuperação por Área

O sistema DEVE calcular recuperação de Cu usando:

- Fórmula linear ( $a \times \text{grade} + b$ ) por área
- OU valor fixo quando especificado

## **REQ-004: Fatores de Mina**

O sistema DEVE aplicar:

- Diluição de mina
- Recuperação de minério (ore recovery)

## **REQ-005: Breakdown Detalhado**

O sistema DEVE fornecer breakdown mostrando:

- Contribuição de cada metal
- Perdas por diluição
- Perdas por recuperação
- Custos detalhados

## **REQ-006: Cenários de Preço**

O sistema DEVE suportar múltiplos cenários de preço:

- Mineral Resources
- Mineral Reserves
- Consensus (Low/Mean/High)

## **REQ-007: Múltiplas Áreas**

O sistema DEVE suportar seleção de:

- Mina (Mine)
- Área dentro da mina

## **REQ-008: Auditabilidade**

O sistema DEVE registrar:

- Todos inputs utilizados
- Fórmula aplicada
- Valores intermediários

## **REQ-009: Unidades Explícitas**

O sistema DEVE exibir unidades para todos valores:

- \$/t ore (NSR por tonelada de minério)
- \$/t conc (Preço por tonelada de concentrado)
- % (percentuais)
- g/t (gramas por tonelada)

## REQ-010: Validação de Inputs

O sistema DEVE validar:

- Teores positivos
- Recovery  $\leq$  100%
- Payability  $\leq$  100%
- Diluição  $\leq$  100%

## 8. Mapeamento para User Stories

Requisito	User Stories Relacionadas
REQ-001	US-03 (Domínio), US-07 (nsr_total)
REQ-002	US-03 (Domínio), US-06 (deductions)
REQ-003	US-04 (payable_metal) - <b>NOVO REQUISITO</b>
REQ-004	US-07 (nsr_total) - <b>EXPANDIR</b>
REQ-005	US-07 (nsr_total), US-13 (Result UI)
REQ-006	US-14 (Scenarios) - <b>EXPANDIR</b>
REQ-007	US-03 (Domínio) - <b>EXPANDIR</b>
REQ-008	US-07 (nsr_total), US-09 (API)
REQ-009	US-13 (Result UI)
REQ-010	US-04, US-09, US-12

## **9. Gaps Identificados nas User Stories**

### **GAP-001: Recuperação Variável por Área**

**Descrição:** User stories assumem recuperação fixa, mas Caraíba usa fórmula linear.

**Ação:** Adicionar suporte a recuperação calculada em US-04.

### **GAP-002: Teores no Concentrado para Subprodutos**

**Descrição:** Au e Ag têm teores no concentrado derivados da razão de concentrado.

**Ação:** Expandir US-05 para calcular teores no concentrado.

### **GAP-003: Múltiplas Áreas por Mina**

**Descrição:** Planilha suporta seleção de Mine + Area.

**Ação:** Expandir modelo de domínio em US-03.

### **GAP-004: Fatores de Mina (Diluição + Ore Recovery)**

**Descrição:** NSR final considera perdas de mina.

**Ação:** Adicionar cálculo em US-07.

### **GAP-005: Price Deck com Múltiplos Cenários**

**Descrição:** Planilha tem cenários pré-definidos (Resources, Reserves, Consensus).

**Ação:** Adicionar seed data em US-03.

## 10. Constantes e Conversões

```
# Conversões de peso
LB_PER_KG = 2.20462
LB_PER_TONNE = 2204.62
KG_PER_TONNE = 1000
GRAM_PER_KG = 1000
GRAM_PER_TONNE = 1_000_000

# Conversões de onça troy
TROY_OZ_PER_GRAM = 0.0321507
GRAM_PER_TROY_OZ = 31.1035

# Conversões de unidade de teor
PCT_TO_FRACTION = 0.01
PPM_TO_FRACTION = 1e-6
GPT_TO_FRACTION = 1e-6 # g/t = ppm para massa
```

## 11. Referência: Planilha Original

### Estrutura de Planilhas

Planilha	Conteúdo
NSR	Calculadora principal, inputs e outputs
Assumptions ->	Navegação (vazia)
Price and Costs	Price deck, termos comerciais, royalties
Processing	Parâmetros de recuperação por área
Data	Lista de minas e áreas

### Células Chave da Planilha NSR

Célula	Conteúdo
E5-F5	Seleção de Mine/Area

Célula	Conteúdo
E10-F12	Head Grades (Cu, Au, Ag)
L7-M10	Conc. Price breakdown
L16-M20	NSR breakdown
Q10-R15	Custos operacionais
U7-U13	Cascata de NSR (Resources → Mine → Processing)