Chassi de Controles Internos

ChassiRO

Manual de uso do pipeline analítico

De CSVs internos a parecer estruturado em uma execução

Bem público colaborativo para o SFN brasileiro

Versão	v0.1.0 (pipeline v5)
Licença	MIT (código) + CC BY 4.0 (catálogo)
Mantenedor	Walter C. Neto
Repositório	github.com/walterCNeto/ChassiRO
API pública	chassiro-api.fly.dev

Contents

1 O que é o ChassiRO

O **ChassiRO** é um catálogo aberto e colaborativo dos controles internos do Sistema Financeiro Nacional brasileiro — mapeando reguladores, normas, processos típicos e taxonomia canônica de riscos. O **pipeline analítico** descrito neste manual transforma esse catálogo em uma ferramenta operacional: ele lê os dados internos da sua instituição (apontamentos, KRIs, autoavaliações), combina com fontes externas públicas (BACEN, CVM, ANBIMA, Procon, mídia), e gera em uma execução um **parecer estruturado de controles internos**, um **nine-box** de impacto × ambiente de controles, e uma **tabela consolidada** de detalhamento por risco.

Para quem é

Profissionais de risco, controles internos, auditoria interna, compliance e governança em instituições financeiras brasileiras — bancos, asset managers, distribuidoras, corretoras, gestoras fiduciárias. Útil também para professores, pesquisadores e quem estuda regulação financeira.

O que ele faz

- Traduz a taxonomia interna de riscos da sua casa para os IDs canônicos do chassi (`RO.U.1` = Risco de Crédito, `RO.U.5` = Risco Cibernético, etc.).
- Calcula **score de controles** (eixo X) combinando sinais internos e externos.
- Calcula **score de impacto** (eixo Y) priorizando a materialidade aprovada pelo CRO da casa.
- Calcula **cobertura** (3ª dimensão) — proporção de riscos declarados nos processos que efetivamente receberam algum sinal nos últimos 12 meses.
- Posiciona cada risco no **nine-box** (3×3 com cores semafóricas: crítico/atenção/ok).
- Escreve um **parecer em Markdown** com sumário executivo, fichas dos riscos críticos, alertas de cobertura e metodologia.

2 Arquitetura do pipeline

O pipeline lê **13 arquivos CSV** de entrada, conecta-se ao **catálogo do chassi** (online ou offline via snapshot), e escreve **3 arquivos de saída**.

Categoria	Arquivos de entrada

Fontes internas	`apontamentos.csv`
	`kris.csv`
	`autoavaliacao.csv`
	`bia_continuidade.csv`
	`auditoria_plano.csv`
	`monitoramento_2linha.csv`

Calibração interna	`materialidade_interna.csv`
	`riscos_declarados.csv`
	`mapa_taxonomia.csv`

Fontes externas (públicas)	`procedentes_bacen.csv`
	`procon_reclamacoes.csv`
	`cvm_sancoes.csv`
	`anbima_sancoes.csv`
	`midia_adversa.csv`
	`reclame_aqui.csv`

Saída	Conteúdo
`output/resultado_consolidado.csv`	Tabela com 1 linha por R0
`output/nine_box.html`	Visualização interativa
`output/parecer.md`	Parecer estruturado

3 Pré-requisitos

3.1 O que você precisa instalado

- **Python 3.10 ou superior**. Para verificar, execute:

```
python --version
```

- **Pacote** `requests` (única dependência externa em modo online; em modo offline o pipeline não usa nada além da biblioteca padrão):

```
python -m pip install requests
```

3.2 O que você precisa baixar

Baixe o pacote demo do repositório oficial:

```
git clone https://github.com/walterCNeto/ChassiRO.git
```

ou baixe o ZIP `ChassiRO-analise-v5.zip` diretamente da página do projeto.

A estrutura esperada do diretório de trabalho é:

```
ChassiRO-analise/
|-- analisar.py             # pipeline
|-- chassi.json             # snapshot do catalogo (modo offline)
|-- README.md
|-- demo/                   # 13 CSVs de entrada
|   |-- apontamentos.csv
|   |-- kris.csv
|   |-- autoavaliacao.csv
|   |-- ...
`-- output/                 # criado na primeira execucao
```

4 Primeira execução (modo demo)

A primeira execução usa os **dados sintéticos** do Banco Modelo S.A. — um banco múltiplo S2 fictício, calibrado com perfil realista. Serve para você ver o pipeline funcionando antes de plugar seus dados reais.

4.1 Modo offline (recomendado para começar)

```
cd ChassiRO-analise

python analisar.py --offline --snapshot chassi.json
```

O modo offline lê o snapshot do catálogo (`chassi.json`, ~181KB) e não exige conexão com a internet. Útil para ambientes corporativos com firewall.

4.2 Modo online (consulta a API ao vivo)

```
python analisar.py
```

O modo online consulta chassiro-api.fly.dev em tempo real. Garante a versão mais recente do catálogo.

> **O que esperar.** Em ~5 segundos o pipeline imprime no console quantas linhas leu de cada CSV, aplica a tradução de taxonomia, conecta no chassi, sobe a hierarquia (`R1.x.y` para `R0.x`) e calcula scores. No final você vê três linhas `==> output/...` indicando os arquivos gerados.

4.3 Conferindo o resultado

Abra os três arquivos gerados:

- `output/nine_box.html` no navegador — você verá o grid 3×3 colorido com os ~39 R0 distribuídos.
- `output/parecer.md` em qualquer editor de Markdown (VSCode, Typora, ou navegador via extensão) — o parecer tem sumário executivo, fichas dos riscos críticos e metodologia.
- `output/resultado_consolidado.csv` no Excel ou equivalente — 1 linha por R0 com ~25 colunas de detalhamento.

5 Plugando os seus dados reais

Esta é a etapa mais trabalhosa, mas é onde o pipeline vira ferramenta operacional. Os CSVs sintéticos servem como template — você substitui o conteúdo mantendo as colunas.

5.1 Etapa 1: traduzir a sua taxonomia

Abra `demo/mapa_taxonomia.csv`. Esse arquivo é o tradutor entre os termos que sua casa usa (em apontamentos, KRIs etc.) e os IDs canônicos do chassi.

Estrutura:

```
termo_interno;risco_id_chassi;confianca;risco_nome_chassi
Credito;R0.U.1;alta;Risco de Credito
Inadimplencia counterparty;R1.U.1.1;alta;Default counterparty
Cibernetico;R0.U.5;alta;Risco Cibernetico
Vazamento de dados;R1.U.5.2;alta;Vazamento e confidencialidade
```

Para cada *termo interno* da sua casa, atribua o `risco_id_chassi` correspondente. Confiança `alta` significa mapeamento direto; `media` indica que pode haver overlap; `baixa` sinaliza ambiguidade.

5.2 Etapa 2: declarar materialidade interna

Em `demo/materialidade_interna.csv` a casa atribui **materialidade aprovada pelo CRO** a cada R0 que considera relevante. Essa materialidade tem peso dobrado no score de impacto e prevalece sobre o default do chassi.

```
risco_id_chassi;score_materialidade;justificativa;aprovado_por;
    data_aprovacao
R0.U.1;5;Carteira de credito = 38% PR. Maior exposicao da casa.;Carlos
    Mendes (CRO);15/01/2026
```

```
R0.U.5;5;Banco digital com PIX 24x7. Vetor critico em 2025/26.;Carlos
    Mendes (CRO);15/01/2026
```

5.3 Etapa 3: declarar riscos por processo

`demo/riscos_declarados.csv` é a planilha que define o **denominador da cobertura**. Cada linha é uma declaração: "no processo P, identificamos que o risco R é relevante".

```
processo_chassi_id;risco_chassi_id;severidade_inerente;declarado_por;
    data_declaracao
P0.B.2;R0.U.1;5;Credito PJ;15/01/2026
P0.B.2;R1.U.1.3;4;Credito PJ;15/01/2026
P0.U.8;R0.U.5;5;Ciberseguranca;15/01/2026
P0.U.8;R1.U.5.4;5;Ciberseguranca;15/01/2026
```

Áreas que não operam um núcleo (ex: seguros num banco múltiplo) **não declaram nada** para esses processos. O pipeline reconhece isso e marca a cobertura como `n/a`.

5.4 Etapa 4: substituir as fontes internas

Substitua o conteúdo dos seguintes CSVs pelos seus dados reais, mantendo as colunas:

- `apontamentos.csv`: apontamentos de auditoria, regulador e gestão. Colunas-chave: `categoria_interna`, `severidade`, `status`, `origem`, `area`.
- `kris.csv`: indicadores chave com 6 meses de histórico (`m1`, `m2`, ..., `m6`).
- `autoavaliacao.csv`: autoavaliação ciclo atual com eficácia e impacto avaliados pelas áreas (1-5).
- `bia_continuidade.csv`: BIA de processos críticos.
- `auditoria_plano.csv`: plano anual de auditoria com status (Concluído / Em execução / Planejado).
- `monitoramento_2linha.csv`: atividades de monitoramento contínuo de Riscos & Controles, Compliance, Privacidade etc.

5.5 Etapa 5: alimentar fontes externas públicas

Aqui está a parte mais original do pipeline: ele agrega **seis fontes públicas** que normalmente ficam em silos diferentes.

5.5.1 BACEN

Busque procedentes pelo CNPJ da instituição em bcb.gov.br/estabilidadefinanceira/rankingreclamacoes. Exporte para `procedentes_bacen.csv`.

5.5.2 CVM SAS

Busque processos administrativos sancionadores em sistemas.cvm.gov.br. Atualize `cvm_sancoes.csv` trimestralmente. Mantenha as colunas: `numero_processo`, `data`, `cnpj_acusado`, `tipo_processo`, `status` (Julgado / Em curso / Encerrado), `categoria_infracao`, `penalidade`.

5.5.3 ANBIMA

Em anbima.com.br/.../orientacoes-e-penalidades estão divulgados **Cartas de Recomendação**, **Termos de Compromisso** e **Julgamentos** desde abril de 2016. Atualize `anbima_sancoes.csv` semestralmente, alinhado com o relatório de supervisão da ANBIMA.

5.5.4 Procon, mídia, Reclame Aqui

`procon_reclamacoes.csv`, `midia_adversa.csv` e `reclame_aqui.csv` têm formatos similares e exigem coleta manual ou via scrapers internos.

> **Importante.** Todas essas fontes são **públicas**. O pipeline foi projetado deliberadamente para usar apenas dados acessíveis a qualquer cidadão — isso permite que outras instituições reproduzam a metodologia sem barreira de acesso.

5.6 Etapa 6: rodar e iterar

```
python analisar.py --offline --snapshot chassi.json
```

Confira o parecer. Se algum R0 ficar classificado de forma contraintuitiva, há três coisas para checar:

1. **Mapeamento da taxonomia**: o termo da sua casa está apontando para o R0 correto?
2. **Materialidade interna**: você aprovou materialidade para esse R0? Se não, o pipeline cai no default do chassi.
3. **Riscos declarados**: o R0 está sendo declarado nos processos relevantes? Se não, a cobertura fica `n/a` e o R0 some da análise de cobertura.

6 Lendo o nine-box

O nine-box é um grid 3×3 com semáforo nas células. Cada bolha representa um R0.

6.1 O que cada eixo significa

- **Eixo X (horizontal): ambiente de controles**. Da esquerda para a direita: vermelho (crítico), amarelo (atenção), verde (ok). Combina apontamentos, KRIs, autoavaliação e **seis fontes externas** (BACEN, CVM SAS, ANBIMA, Procon, mídia, Reclame Aqui).
- **Eixo Y (vertical): impacto inerente**. De cima para baixo: alto, médio, baixo. Combina materialidade interna (peso dobrado), criticidade BIA e impacto autoavaliado.

6.2 O que o tamanho da bolha significa

A 3ª dimensão do modelo. O tamanho reflete **cobertura**:

- **Bolha pequena**: cobertura alta (≥70%) — a maior parte dos riscos declarados nos processos do banco recebeu algum sinal nos últimos 12 meses. Ambiente validado.
- **Bolha média**: cobertura entre 40% e 70%.
- **Bolha grande**: cobertura baixa (<40%) — vários riscos declarados sem nenhum sinal. Ponto cego.

- **Bolha em itálico discreto**: cobertura `n/a` — o núcleo não opera no banco (ex: seguros num banco múltiplo).

6.3 O quadrante mais importante

Coluna esquerda inteira é vermelha. Mas dentro dela há três níveis de saturação — linha do alto é o mais crítico (alto impacto × controles ruins), exigindo ação imediata. A mesma lógica vale para a coluna verde: alto impacto × controles bons é uma posição diferente de baixo impacto × controles bons.

7 Lendo o parecer

O parecer em Markdown segue uma estrutura padrão:

- **Sumário executivo**: contadores de quadrante e de cobertura.
- **Riscos em quadrante crítico**: ficha individual de cada R0 alto × vermelho. Lista áreas envolvidas, apontamentos, KRIs, sinais externos, scores e justificativas detalhadas.
- **Riscos em zona de atenção**: tabela compacta dos R0 alto × amarelo ou médio × vermelho.
- **Riscos em zona confortável**: lista resumida.
- **Alertas de cobertura**: R0 com riscos declarados sem nenhum sinal — pontos cegos do ambiente de controles.
- **Recomendações**: ações por quadrante.
- **Metodologia**: pesos e fórmulas.

8 Cadência de uso recomendada

Frequência	Atividade
Trimestral	Atualizar BACEN procedentes, CVM SAS, mídia adversa, Reclame Aqui, apontamentos, KRIs. Rodar pipeline e levar parecer ao Comitê de Risco Operacional.
Semestral	Atualizar ANBIMA (alinha com relatório de supervisão ANBIMA), revisar plano anual de auditoria, recalibrar materialidade interna se necessário.
Anual	Revisar mapeamento de taxonomia (novos produtos, novos riscos), revisar declaração de riscos por processo, atualizar BIA.
Quando houver evento	Atualizar imediatamente após qualquer sanção regulatória, evento operacional relevante ou mudança organizacional.

9 Modos de uso avançado

9.1 Apenas algumas fontes externas

Se sua casa não tem volume relevante em alguma fonte (ex: não tem operação de mercado de capitais e portanto CVM/ANBIMA não se aplicam), basta deixar o CSV correspondente vazio (apenas a linha de cabeçalho). O pipeline tolera planilhas ausentes ou vazias.

9.2 Snapshot do catálogo congelado

Para auditoria e reprodutibilidade, mantenha uma cópia do `chassi.json` junto com cada execução. Assim você pode reproduzir o resultado meses depois mesmo que o catálogo público tenha evoluído:

```
python analisar.py --offline --snapshot chassi-2026-Q1.json
```

9.3 Ajustar pesos das fontes

Os pesos das fontes externas são parâmetros explícitos na função `calcular_score_controles()` do `analisar.py`. Para a versão demo o BACEN tem peso 0.5 por procedente (cap 6.0), CVM julgado tem 2.0 (cap 6.0), ANBIMA julgamento 1.5 (cap 3.0). Você pode ajustar esses pesos para refletir o perfil de risco da sua casa — mas documente a calibração na metodologia da casa.

10 Limitações conhecidas

- O catálogo cobre os principais reguladores e normas mas **não é exaustivo**. Versão atual tem 28 normas e 90 riscos R0/R1.
- Cada execução é **snapshot do momento**. Não há ainda camada nativa de série histórica; recomenda-se versionar os outputs em `output/historico/AAAA-Qn/`.
- As fontes externas são públicas e portanto têm **latência**. CVM SAS pode demorar meses entre instauração e julgamento; ANBIMA divulga semestralmente.

11 Onde pedir ajuda

- **Issues do GitHub**: github.com/walterCNeto/ChassiRO/issues — bugs, dúvidas técnicas, sugestões de novas normas.
- **Pull requests**: contribuições ao catálogo são bem-vindas, especialmente novos riscos R1, novas normas e novos vínculos processo×risco.
- **Landing**: waltercneto.github.io/ChassiRO.

O ChassiRO é um bem público colaborativo. Código sob licença MIT, catálogo sob CC BY 4.0. Use, adapte e contribua.