

PRÉ-REQUISITOS · MIGRAÇÃO

Pré-requisitos: Migração Bitbucket Cloud para GitLab™ self-managed

Documento enviado após a escolha do serviço PS-MIG-04: o que o cliente provisiona, as credenciais que entrega à Pointer, as informações a enviar e o checklist de prontidão.

CÓDIGO**PR-MIG-04****VERSÃO****1.2****TIPO****Pré-requisitos****DATA****1 de outubro de 2026**

O3S Consultoria e Tecnologia da Informação Ltda.
contato@pointertech.digital · +55 (61) 99874-4648 · pointertech.digital

GITLAB is a trademark of GitLab Inc. in the United States and other countries and regions.

**Select
Partner**

PRÉ-REQUISITOS • PR-MIG-04 • OFERTA PS-MIG-04

Pré-requisitos: Migração Bitbucket Cloud para GitLab self-managed

Documento enviado ao cliente depois da escolha do serviço **PS-MIG-04 — Migração Bitbucket Cloud para GitLab self-managed**. Lista o que o cliente provisiona, as credenciais que entrega, as informações a enviar à Pointer, o checklist de prontidão e a matriz do que é migrado. O prazo de devolução das informações e do checklist é definido no kick-off.

Os itens do checklist marcados como **Preflight automático** são conferidos pelo pipeline Pointer contra a instância de destino real antes da primeira onda; os demais são conferidos na reunião de prontidão ou pelo próprio cliente. Nenhuma onda é executada com erro no preflight.

Conteúdo

Origem e destino suportados

Configuração da instância de destino

Workspace e repositórios no Bitbucket Cloud

Host de migração

Runner do engajamento

Rede

Usuários e contribuições

Governança da migração

Critérios de aceite por onda

Entrega das credenciais

Credenciais que o cliente entrega à Pointer

Informações a enviar

Matriz de itens migrados

Checklist de prontidão

1. Origem e destino suportados

Item	Suportado
Origem	Bitbucket Cloud (bitbucket.org), um workspace por plano de migração

Item	Suportado
Destino	Somente GitLab self-managed do cliente, inclusive instância implantada pela Pointer
Método	Importador Bitbucket Cloud da instância de destino, disparado pelo GitLab Congregate, seguido das etapas pós-migração do GitLab Congregate. A instância de destino busca os dados no Bitbucket Cloud
Versão do destino	18.9 ou superior: a importação usa API token da Atlassian, aceito pelo importador desde a versão 18.9; o suporte a app passwords do Bitbucket Cloud foi removido na versão 19.0 (documentação GitLab)
Estrutura no destino	Grupo pai + chave do projeto no Bitbucket + repositório
Itens migrados pelo importador	Repositórios, branches, pull requests com comentários e labels
Sob consulta	Issues, wiki, milestones, LFS e membros: escopo e condições definidos em proposta específica, com execução piloto
Destinos e origens fora do escopo	GitLab.com e GitLab Dedicated como destino; Bitbucket Server/Data Center e AWS CodeCommit como origem: roadmap

- Com origem Bitbucket Cloud não há arquivamento da origem, reescrita de referências nos repositórios nem contagem independente origem × destino; o pipeline verifica cada importação pelo status e pelas estatísticas da importação.

2. Configuração da instância de destino

Item	Especificação	Obrigatório
Integração Bitbucket Cloud	Habilitada na instância (documentação GitLab: <i>You must enable the Bitbucket Cloud integration</i>)	Sim
Fonte de importação Bitbucket Cloud	Habilitada nas configurações de importação e exportação da instância (<i>Admin → Settings → General → Import and export settings</i> , fontes de importação). Em GitLab self-managed nenhuma fonte de importação vem habilitada por padrão. O preflight reprova se ausente	Sim
Grupo pai	Grupo existente, privado ou interno, que recebe a migração. O preflight reprova grupo inexistente ou público	Sim
Versão	18.9 ou superior	Sim

3. Workspace e repositórios no Bitbucket Cloud

- O workspace é informado no plano de migração e define o escopo da listagem.

- Pull requests de fork ou entre projetos diferentes viram merge requests vazios (documentação GitLab).
- Revisores dos pull requests são importados quando o username casa com uma identidade Bitbucket vinculada na instância de destino.
- Issues importadas recebem label pelo tipo (bug, enhancement, proposal, task); estados resolved, invalid, duplicate, wontfix e closed viram issue fechada.
- Imagens e anexos inline em Markdown continuam como links para o Bitbucket Cloud e deixam de funcionar se o repositório de origem for apagado.
- Durante a janela da onda, os repositórios da onda não recebem alterações no Bitbucket Cloud.

4. Host de migração

Item	Especificação	Obrigatório
Máquina	Linux x86_64 dedicada ao engajamento, sem outros serviços: o usuário do runner participa do grupo docker , o que equivale a root no host. Um host por cliente	Sim
Recursos de partida	4 vCPU, 16 GB de RAM e 100 GB de disco SSD (imagem do GitLab Congregate com cerca de 3,3 GB; o banco MongoDB da ferramenta cresce com a listagem da origem). A documentação do GitLab Congregate (página <i>Migration Workflows</i>) pede, no mínimo, ambiente com Docker e 4 CPU, 8 GB de RAM e 20 GB de disco	Sim
Contêineres	Docker Engine e Docker Compose v2; a ausência reprovava o início de cada job	Sim
Swap	2 GB de swap, conforme a documentação do GitLab Congregate	Não
Certificados	Certificado válido no destino. CA corporativa em arquivo no host (arquivo inexistente reprovava o job), somada às raízes do contêiner do GitLab Congregate	Condicional: CA interna
Proxy	Proxy de saída informado no plano de migração e configurado também no daemon Docker	Condicional: proxy corporativo
Acompanhamento	A interface do GitLab Congregate e o Flower ficam só em 127.0.0.1 do host; o acompanhamento é por túnel SSH (usuário@host)	Não

- No encerramento, o pipeline remove a stack e todos os dados do engajamento no host (listagens, logs e resultados).

5. Runner do engajamento

Item	Especificação	Obrigatório
GitLab Runner	Instalado no host de migração, executor shell, registrado no projeto do engajamento na GitLab.com da Pointer com o token de registro criado nesse projeto	Sim
Configuração	Runner de projeto com a tag do engajamento; <i>Run untagged jobs</i> desmarcado; <i>Protected</i> e <i>Lock to current projects</i> marcados; tempo máximo de job de 8 horas ou mais	Sim
Usuário do runner	<code>gitlab-runner</code> no grupo <code>docker</code> ; em Ubuntu, remover <code>/home/gitlab-runner/.bash_logout</code> (o <code>clear_console</code> encerra jobs shell)	Sim
Encerramento	Runner removido do projeto e desinstalado do host	Sim

6. Rede

Conexão	Protocolo e observação	Obrigatório
Host de migração → Bitbucket Cloud	HTTPS; API em <code>https://api.bitbucket.org</code>	Sim
Host de migração → instância de destino	HTTPS	Sim
Host de migração → gitlab.com	HTTPS: runner e projeto do engajamento	Sim
Host de migração → registry.gitlab.com	HTTPS: imagens do GitLab Congregate e da plataforma Pointer. Para pull de registry.gitlab.com, a documentação da GitLab.com lista também <code>*.storage.googleapis.com</code> e <code>*.cdn.registry.gitlab-static.net</code>	Sim
Host de migração → Docker Hub	HTTPS: imagens do MongoDB e do Redis	Sim
Instância de destino → Bitbucket Cloud	HTTPS: o importador Bitbucket Cloud é executado pela instância de destino	Sim
Estação do engenheiro → host de migração	SSH, para o túnel de acompanhamento	Não

- O host de migração só faz conexões de saída. Espelhos internos equivalentes podem substituir gitlab.com, registry.gitlab.com e Docker Hub.
- Operação sem acesso à internet (air-gapped) está no roadmap.

7. Usuários e contribuições

- Não há usuários placeholder nem reatribuição depois da importação: o mapeamento posterior à migração não se aplica ao Bitbucket Cloud (documentação GitLab).
- O importador usa o nickname do Bitbucket e procura a identidade Bitbucket vinculada na instância de destino; sem correspondência, o autor passa a ser a conta que fez a importação, com referência ao autor original.
- Para a correspondência, cada usuário ajusta o nome público da conta Atlassian igual ao username do Bitbucket e conecta a conta Bitbucket no perfil da instância de destino (*Service sign-in*), antes da onda. A atribuição de autoria por identidade vinculada está no roadmap.
- O pipeline não cria usuários a partir do Bitbucket Cloud (o GitLab Congregate não migra contas de usuário nem chaves SSH do Bitbucket Cloud): os usuários do destino vêm de LDAP, SAML ou cadastro.
- A documentação do GitLab Congregate descreve a criação, pelo provedor de identidade, de um usuário de sistema para receber as contribuições órfãs; o pipeline não exige essa conta.
- A conta dona do token do destino é a conta da importação.

8. Governança da migração

- Aprovador de cada onda e do rollback, do lado do cliente.
- Janela de manutenção de cada onda e congelamento de escrita no Bitbucket Cloud durante a janela.
- Decisão sobre a conta que recebe as contribuições sem correspondência (conta dona do token do destino).
- Comunicação aos usuários sobre o fim do uso dos repositórios no Bitbucket Cloud (o pipeline não arquiva a origem Bitbucket Cloud).
- Responsável por cada item de tratamento manual da matriz, inclusive a conversão dos Bitbucket Pipelines.

9. Critérios de aceite por onda

- Relatório da onda sem erro: todo repositório existe no destino e nenhuma importação terminou com falha.
- Alertas (relações com falha, diferença entre objetos lidos e importados) analisados e aceitos ou corrigidos, com registro no registro de decisões.
- Validação funcional pelo cliente nos repositórios críticos da onda: clone, merge requests e, quando usados, issues e wiki.

10. Entrega das credenciais

Cada credencial é cadastrada como variável de CI/CD do projeto do engajamento, com as opções *Protected* e *Masked and hidden* e escopo restrito ao ambiente de migração, que é protegido e exige aprovação para a migração e o rollback de cada onda. O e-mail da conta Atlassian também fica em variável do ambiente, fora do repositório do engajamento. O pipeline lista, sem valores, as variáveis exigidas pelo plano, e o preflight reprova se faltar alguma. A configuração do GitLab Congregate com os tokens só existe durante cada job no host de migração.

Tokens nunca são colados em chat, issue ou arquivo; se isso ocorrer, o token é revogado e outro é emitido. No encerramento, os tokens usados na origem e no destino são revogados e as contas de administrador criadas para o engajamento são desativadas.

11. Credenciais que o cliente entrega à Pointer

Nenhum valor secreto é enviado por e-mail, chat, issue ou arquivo. Cada credencial é cadastrada como variável de CI/CD protegida no projeto do engajamento, com escopo do ambiente, e revogada no encerramento.

Credencial	Finalidade	Permissões	Formato e validade	Obrigatório
Token de acesso pessoal da instância de destino	Usado pelo GitLab Congregate (disparo do importador e etapas pós-migração), pela verificação das importações e pelo rollback.	Escopo <code>api</code> , e também <code>admin_mode</code> quando o Admin Mode estiver habilitado na instância. Usuário administrador da instância: sem administrador o preflight alerta e não confere as configurações de importação. A importação exige, no mínimo, papel Maintainer ou Owner no grupo de destino (documentação GitLab).	Personal access token da instância GitLab de destino, cadastrado como variável de CI/CD <code>PS_MIG_DESTINO_TOKEN</code> (Protected, Masked and hidden, escopo do ambiente de migração) Expiração na data estimada da última onda, conforme a documentação do GitLab Congregate; o preflight alerta com menos de 7 dias. Revogado no encerramento.	Sim

Credencial	Finalidade	Permissões	Formato e validade	Obrigatório
API token da Atlassian (Bitbucket)	Usado pelo GitLab Congregate na listagem do workspace e repassado ao importador Bitbucket Cloud da instância de destino.	API token com escopos (<i>API token with scopes</i>), aplicativo Bitbucket, com <code>read:workspace:bitbucket</code> , <code>read:project:bitbucket</code> , <code>read:repository:bitbucket</code> , <code>read:pullrequest:bitbucket</code> , <code>read:issue:bitbucket</code> , <code>read:wiki:bitbucket</code> e <code>read:user:bitbucket</code> . App passwords não funcionam.	API token da Atlassian, cadastrado como variável de CI/CD <code>PS_MIG_ORIGEM_TOKEN</code> (Protected, Masked and hidden, escopo do ambiente de migração) Expiração na data estimada da última onda. Revogado no encerramento.	Sim
E-mail da conta Atlassian dona do API token	Autenticação no Bitbucket Cloud junto com o API token, na listagem e no importador.	Mesma conta do API token.	Texto, cadastrado como variável do ambiente de migração <code>PS_MIG_ORIGEM_USUARIO</code> (forá do repositório, por ser dado pessoal) Até o encerramento	Sim

12. Informações a enviar

Informações sem sigilo, devolvidas pelo cliente antes da reunião de prontidão. Os campos marcados como obrigatórios são conferidos pela validação automática do pipeline.

Origem

Informação	Exemplo	Obrigatório	Observação
Workspace do Bitbucket Cloud	orgao-exemplo	Sim	Define o escopo da listagem.
Projetos do Bitbucket no escopo (chave)	SIS, PORTAL	Sim	
Uso de issues e wiki no Bitbucket Cloud	Wiki em 3 repositórios; sem issues	Sim	Issues e wiki: sob consulta; escopo e condições definidos em proposta específica, com execução piloto.

Destino

Informação	Exemplo	Obrigatório	Observação
URL da instância de destino	https://gitlab.exemplo.gov.br	Sim	
Versão e nível de assinatura do destino	19.4.1, Premium	Sim	18.9 ou superior.
Grupo pai no destino	migrados-bitbucket	Sim	Grupo privado ou interno existente.

Host de migração e rede

Informação	Exemplo	Obrigatório	Observação
Nome e sistema operacional do host de migração	host-migracao.exemplo.gov.br, Ubuntu 24.04	Sim	
Proxy de saída	https://proxy.exemplo.gov.br:3128; exceções: .exemplo.gov.br	Condicional: proxy corporativo	HTTPS, HTTP e exceções.
Caminho da CA corporativa no host	/etc/pki/ca-trust/source/anchors/ac-exemplo.pem	Condicional: CA interna	Caminho absoluto de arquivo existente no host.
Acesso SSH para acompanhamento	usuario@host-migracao.exemplo.gov.br	Não	Túnel para a interface do GitLab Congregate.
Confirmação de conectividade destino → Bitbucket Cloud por HTTPS	Sim	Sim	

Usuários

Informação	Exemplo	Obrigatório	Observação
Origem dos usuários no destino	LDAP	Sim	LDAP, SAML ou cadastro.
Usuários que vinculam a identidade Bitbucket antes da onda	Não	Não	Sem vínculo, a autoria fica na conta da importação.

Ondas

Informação	Exemplo	Obrigatório	Observação
Repositórios de cada onda	piloto: SIS/app-bb, SIS/lib-bb	Sim	Caminho chave do projeto/repositório; o mesmo repositório não se repete entre ondas. Também por planilha (colunas onda e projeto).
Nome, data e descrição de cada onda	piloto; 10/11/2026; repositórios de baixo risco	Sim	A primeira onda é a piloto.
Janela de manutenção de cada onda	10/11/2026, 19h às 23h	Sim	
Aprovador de cada onda e do rollback	Nome e cargo	Sim	
Janela de rollback	24 horas	Não	De 1 a 720 horas; padrão 24.
Confirmação do congelamento de escrita no Bitbucket Cloud durante a janela	Sim	Sim	

Pós-migração

Informação	Exemplo	Obrigatório	Observação
Responsáveis pelos itens de tratamento manual	Bitbucket Pipelines e permissões de branch: equipe de plataforma	Sim	Conforme a matriz.

Contatos

Informação	Exemplo	Obrigatório	Observação
Patrocinador	Nome, e-mail, telefone	Sim	
Ponto focal técnico	Nome, e-mail, telefone	Sim	
Administrador do workspace no Bitbucket Cloud	Nome, e-mail	Sim	
Administrador da instância de destino	Nome, e-mail	Sim	
Infraestrutura, rede e segurança	Nome, e-mail	Sim	Host, firewall, proxy e certificados.

13. Matriz de itens migrados

- Nativo** Migrado pelo método oficial (importador Bitbucket Cloud da instância de destino), disparado pelo pipeline
- Pipeline Pointer** Tratado pelo pipeline Pointer após a importação (etapas pós-migração do GitLab Congregate)
- Ajuste manual** Exige ajuste manual após a migração (com responsável definido no plano)
- Não migra** Não é migrado; recriação fora do escopo, salvo acordo

O tratamento de cada item segue a documentação oficial do método de migração e é conferido na onda piloto do engajamento.

Repositório

Item	Tratamento	Observação
Repositório Git (branches, tags)	Nativo	
Descrição do repositório	Nativo	
Objetos LFS	Nativo	
Wiki	Nativo	
Permissões de branch	Ajuste manual	O pós-migração do GitLab Congregate para Bitbucket Cloud não migra permissões de branch; a matriz do GitLab Congregate lista o mapeamento como suportado sem distinguir Bitbucket Server e Cloud. Adotado o código do GitLab Congregate 8.6.0.

Item	Tratamento	Observação
Estado arquivado do repositório	Pipeline Pointer	O GitLab Congregate arquiva no destino o repositório que está arquivado na origem.

Pull requests → merge requests

Item	Tratamento	Observação
Pull requests e comentários	Nativo	Estados aberto, fechado e mergeado; revisores quando o username casa com identidade Bitbucket vinculada; pull request de fork ou entre projetos diferentes vira merge request vazio.
Aprovações de pull request	Não migra	Não importadas (documentação GitLab e matriz do GitLab Congregate).

Issues e planejamento

Item	Tratamento	Observação
Issues e comentários	Nativo	Label pelo tipo (bug, enhancement, proposal, task); estados resolved, invalid, duplicate, wontfix e closed viram issue fechada.
Milestones	Nativo	
Labels	Nativo	

CI/CD e automação

Item	Tratamento	Observação
Bitbucket Pipelines	Ajuste manual	Conversão para GitLab CI/CD, fora da importação.

Usuários e acesso

Item	Tratamento	Observação
Membros do repositório	Pipeline Pointer	Etapa fixa do pós-migração do GitLab Congregate: adiciona os membros ao projeto de destino.
Contas de usuário e chaves SSH	Não migra	O GitLab Congregate não migra contas de usuário nem chaves SSH do Bitbucket Cloud, por falta de suporte na API do Bitbucket Cloud.

Conteúdo não migrado

Item	Tratamento	Observação
Imagens e anexos inline em Markdown	Não migra	Permanecem como links externos para o Bitbucket Cloud e deixam de funcionar se o repositório de origem for apagado (documentação GitLab).

Notas da matriz

- **A confirmar na onda piloto:** regras de aprovação (não importadas pelo importador nem pelo GitLab Congregate, segundo a documentação GitLab e a matriz do GitLab Congregate); webhooks (não suportados pelo importador nem pelo GitLab Congregate, segundo a matriz do GitLab Congregate).
- Autoria: sem identidade Bitbucket vinculada ao usuário do destino, pull requests e comentários ficam na conta da importação.

14. Checklist de prontidão

Itens conferidos antes da execução. "Preflight automático" indica verificação feita pelo pipeline contra o ambiente real; os demais são conferidos na reunião de prontidão ou pelo cliente.

☐ Item	Verificado por
☐ Docker Engine e Docker Compose v2 presentes no host; CA informada existente no host (verificado na preparação do host).	Preflight automático
☐ Variáveis de CI/CD exigidas pelo plano cadastradas, inclusive o e-mail da conta Atlassian.	Preflight automático
☐ Instância de destino acessível e token do destino válido, com papel de administrador.	Preflight automático
☐ Token do destino com validade maior que 7 dias.	Preflight automático
☐ Fonte de importação Bitbucket Cloud habilitada no destino.	Preflight automático
☐ Grupo pai no destino existente e privado ou interno.	Preflight automático
☐ Configuração do GitLab Congregate validada.	Preflight automático
☐ Integração Bitbucket Cloud habilitada no destino e versão 18.9 ou superior.	Reunião de prontidão
☐ Instância de destino alcança o Bitbucket Cloud por HTTPS.	Reunião de prontidão
☐ API token da Atlassian com os sete escopos de leitura exigidos.	Reunião de prontidão
☐ Runner online, com a tag do engajamento, protegido, travado no projeto e com tempo máximo de job de 8 horas ou mais.	Reunião de prontidão
☐ Plano de ondas com onda piloto, janelas, aprovadores e janela de rollback definidos.	Reunião de prontidão
☐ Host de migração provisionado conforme a especificação, com saída HTTPS para Bitbucket Cloud, destino, gitlab.com, registry.gitlab.com e Docker Hub.	Cliente
☐ API token, e-mail da conta Atlassian e token do destino cadastrados como variáveis protegidas.	Cliente
☐ Congelamento de escrita comunicado aos usuários do Bitbucket Cloud para a janela de cada onda.	Cliente