

PRÉ-REQUISITOS · MIGRAÇÃO

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

Documento enviado após a escolha do serviço PS-MIG-01: o que o cliente provisiona, as credenciais que entrega à Pointer, as informações a enviar e o checklist de prontidão.

CÓDIGO**PR-MIG-01****VERSÃO****1.2****TIPO****Pré-requisitos****DATA****1 de outubro de 2026**

Q3S 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-01 · OFERTA PS-MIG-01

Pré-requisitos: Migração GitLab para GitLab self-managed

Documento enviado ao cliente depois da escolha do serviço **PS-MIG-01 — Migração GitLab 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 origem e o destino reais 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

Preparação da instância de origem

Host de migração

Runner do engajamento

Rede

Usuários e contribuições

Governança da migração

Critérios de aceite por onda

Método por arquivo: diferenças em relação ao Direct Transfer

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	GitLab.com; GitLab self-managed
Destino	Somente GitLab self-managed do cliente, inclusive instância implantada pela Pointer
Método padrão	Direct Transfer orquestrado pelo GitLab Congregate
Versões no Direct Transfer	Origem e destino na versão 16.8 ou superior; origem no máximo duas versões minor anteriores ao destino. Fora da regra o preflight reprova; origem mais nova que o destino gera alerta
Método alternativo por arquivo	Exportação e importação de projetos, quando o Direct Transfer não é possível (rede ou versão). Destino na mesma versão da origem ou posterior; a instância de destino importa exportações de até duas versões minor anteriores. Exige token de administrador na origem e no destino
Nível de assinatura	Recursos pagos (por exemplo épicos, regras de aprovação, push rules, wiki de grupo) só migram com o nível correspondente no destino; o nível da origem é informado no plano
Destinos fora do escopo	GitLab.com e GitLab Dedicated como destino: roadmap

Item	Suportado
Outras origens	GitHub (PS-MIG-02), Azure DevOps (PS-MIG-03), Bitbucket Cloud (PS-MIG-04); Bitbucket Server/Data Center e AWS CodeCommit: roadmap

- Documentação GitLab: a migração de projetos por Direct Transfer está em disponibilidade geral desde a versão 18.3; o mapeamento de contribuições para usuários placeholder, desde a versão 18.4 (habilitado em GitLab self-managed a partir da 17.7).
- O runbook do GitLab Congregate exige destino GitLab self-managed 17.7 ou superior, por causa de um defeito anterior que impedia atualizar permissões depois da redistribuição de usuários.
- O preflight confere as versões só no Direct Transfer; no método por arquivo a regra de versões é conferida na reunião de prontidão.

2. Configuração da instância de destino

Item	Especificação	Obrigatório
Direct Transfer habilitado	<i>Admin → Settings → General → Import and export settings → Allow migrating GitLab groups and projects by direct transfer.</i> O preflight reprova se estiver desligado	Sim (Direct Transfer)
Grupo pai	Grupo existente, privado ou interno, que recebe a migração. Destino de cada item = grupo pai + caminho do item na origem. O preflight reprova grupo inexistente ou público	Sim
Redistribuição sem confirmação de cada usuário	<i>Skip confirmation when administrators reassign placeholder users;</i> decisão do administrador da instância. Ligada, a redistribuição em lote é imediata; desligada, cada usuário aceita o pedido de redistribuição antes de contribuições e memberships aparecerem. O preflight alerta se estiver desligada com a redistribuição no plano	Condicional: redistribuição em lote no plano
Sidekiq e espaço temporário	Sidekiq configurado para importações; espaço livre em <code>/tmp</code> para criar e extrair os arquivos da migração (documentação GitLab)	Sim (Direct Transfer)
Object storage	Itens da origem em object storage: <code>proxy_download</code> habilitado na origem ou acesso da instância de destino ao object storage da origem (documentação GitLab)	Condicional: origem com object storage
Exclusão imediata no destino	Permissão de exclusão imediata de grupos e projetos no destino, para rollback com exclusão permanente e remigração no mesmo caminho (preparativos da documentação do GitLab Congregate)	Condicional: rollback permanente
Registry de destino	Endereço do container registry de destino e certificado confiável pelo Docker do host de migração	Condicional: cópia de imagens
Nível de assinatura	Nível que cubra os recursos pagos usados na origem	Condicional: recursos pagos na origem

- Método por arquivo: a fonte de importação exigida no destino não é conferida pelo preflight; conferência na reunião de prontidão.
- Rate limits da origem e do destino ajustados para a migração, inclusive temporariamente, e, opcionalmente, ajustes de memória e filas do Sidekiq no destino constam dos preparativos da documentação do GitLab Congregate.

3. Preparação da instância de origem

- Origem GitLab self-managed: Direct Transfer habilitado também na origem por um administrador. A documentação GitLab exige a opção nas duas instâncias; o preflight confere só o destino.
- *Default minimum role required to create projects* diferente de *No one*: com *No one*, grupos com projetos não são importados (documentação GitLab).
- Snippets habilitados no projeto de origem para que os snippets sejam importados.
- Espaço livre em `/tmp` na origem self-managed e, com object storage, `proxy_download` habilitado.
- Nível de assinatura da origem (Free, Premium ou Ultimate) informado no formulário.
- Preparativos da documentação do GitLab Congregate: limpar repositórios (exportação abaixo de 5 GB por projeto) e tags de registry sem uso; consolidar usuários (remover, pular ou migrar inativos); congelamento de escrita comunicado por broadcast message e, opcionalmente, modo de manutenção.

4. Host de migração

Item	Especificação	Obrigatório
Máquina	Linux x86_64 dedicada ao engajamento, sem outros serviços: o usuário do runner participa do grupo <code>docker</code> , 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 reprova o início de cada job	Sim
Swap	2 GB de swap, conforme a documentação do GitLab Congregate	Não
Disco para imagens	Espaço para a maior imagem de contêiner da origem (cópia tag a tag, com remoção local depois de cada envio)	Condicional: cópia de imagens
Certificados	Certificados válidos na origem e no destino. CA corporativa em arquivo no host (arquivo inexistente reprova o job), somada às raízes do contêiner do GitLab Congregate	Condicional: CA interna
CA para cópia de imagens	Docker do host confiando na CA do registry e da instância GitLab de destino, instalada no repositório de certificados do sistema (<code>update-ca-certificates</code> ou <code>update-ca-trust</code>) e Docker reiniciado; só <code>/etc/docker/certs.d/<registry>/ca.crt</code> não basta, porque o envio autentica em <code>https://<instância de destino>/jwt/auth</code>	Condicional: cópia de imagens com 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 <code>127.0.0.1</code> do host; o acompanhamento é por túnel SSH (usuário@host)	Não

- A documentação do GitLab Congregate orienta dimensionar o disco para as exportações de grupos e projetos e para o pull das imagens de registry.
- 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 → instância de origem	HTTPS (porta da origem)	Sim
Host de migração → instância de destino	HTTPS	Sim
Host de migração → gitlab.com	HTTPS: runner e projeto do engajamento	Sim

Conexão	Protocolo e observação	Obrigatório
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 *.storage.googleapis.com e *.cdn.registry.gitlab-static.net	Sim
Host de migração → Docker Hub	HTTPS: imagens do MongoDB e do Redis	Sim
Instância de destino → instância de origem	HTTPS. O Direct Transfer é executado pela instância de destino e exige HTTPS; firewalls sem bloqueio entre origem e destino	Sim (Direct Transfer); não no método por arquivo
Host de migração → registries de origem e de destino	Porta de cada registry	Condicional: cópia de imagens
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.
- No método por arquivo basta o host alcançar a origem e o destino.
- Operação sem acesso à internet (air-gapped) está no roadmap.

7. Usuários e contribuições

- Direct Transfer: usuários nunca são criados pela migração; contribuições e memberships vão para usuários placeholder mesmo quando existe no destino um usuário com o mesmo e-mail (documentação GitLab).
- Criação de usuários pelo GitLab Congregate (opcional): exige token de administrador na origem e no destino; o administrador da origem é criado como `<usuário>_migrated` quando o nome já existe.
- Usuários já provisionados no destino (LDAP ou SAML): só a reatribuição em lote, sem criação.
- Correspondência origem → destino: por e-mail (padrão, pela API de administrador da origem), por username ou por planilha origem → destino, que vale antes das outras duas. Reatribuição com origem GitLab.com: sob consulta; escopo e condições definidos em proposta específica, com execução piloto.
- Reatribuição dos usuários placeholder: pedida pelo Owner do grupo de topo; com a opção de reatribuição sem confirmação ligada no destino, é imediata.
- Método por arquivo: não há usuários placeholder. Autoria e memberships são mapeados na importação pelo e-mail público da origem igual ao e-mail primário no destino; usuário inexistente no destino nesse momento fica com a autoria na conta da importação, sem reatribuição depois. Por isso os usuários são criados ou provisionados antes da primeira onda. Membros com papel Owner são importados como Maintainer.
- A conta dona do token do destino é a conta da importação: recebe as contribuições sem usuário correspondente.

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 na origem durante a janela.
- Responsável pelos usuários placeholder sem correspondente (Owner do grupo de topo) e prazo.
- Responsável pela revisão e aprovação dos merge requests de reescrita de referências.
- Momento do corte (arquivamento da origem) e comunicação aos usuários.
- Responsável por cada item de tratamento manual da matriz.

9. Critérios de aceite por onda

- Relatório da onda sem erro: todo item existe no destino, número de projetos igual ao da origem. Direct Transfer **finished**, branches, tags, issues e merge requests iguais na contagem independente.

- Falhas de relação e divergências de contagem (alertas) analisadas e aceitas ou corrigidas, com registro no registro de decisões. Sem usuários no destino, é esperado que regras de aprovação fiquem sem aprovadores e que contribuições e memberships fiquem em usuários placeholder.
- Com reatribuição: todos os usuários reatribuídos (ou aguardando aceite, se o destino exigir confirmação), aprovadores das regras reaplicados e autoria em placeholder e membros iguais na contagem.
- Bytes de LFS menores logo após a migração podem ser atraso das estatísticas do destino: nova conferência antes de abrir chamado.
- Validação funcional pelo cliente nos projetos críticos da onda: clone, merge requests, issues e pipelines.

10. Método por arquivo: diferenças em relação ao Direct Transfer

- Merge requests: só o último diff é preservado; só o último pipeline do merge request fica visível.
- Pipelines importados ficam arquivados; agendamentos chegam inativos e atribuídos ao usuário que fez a importação.
- Membros com papel Owner são importados como Maintainer. Membros herdados viram membros diretos quando a exportação é feita pelo Owner do grupo de topo; com o grupo de topo não migrado, a importação por arquivo adiciona como membros diretos do projeto os membros herdados desse grupo.
- Níveis de acesso de branches e tags protegidas voltam para Maintainer, salvo quando quem importa é Owner do grupo de topo ou administrador.
- Não exportados: tokens de trigger, logs e artefatos de jobs, container registry, variáveis de CI/CD, allowlist do job token, webhooks, tokens criptografados, número de aprovações exigidas, limites de tamanho de repositório, deploy keys com push em branch protegida, secure files, políticas de segurança, links entre issues e itens vinculados, links para merge requests relacionados e variáveis de agendamentos.
- Regras de aprovação: só *Approvals for protected branches* e *Eligible approvers*.
- Deploy keys não são importadas pela exportação; o pós-importação do GitLab Congregate as recria.
- Exportação por arquivo de grupos está descontinuada pela GitLab. No pipeline, onda só de subgrupos roda com opção própria, e onda que mistura grupo de topo e subgrupo solto é recusada: grupo de topo e subgrupo avulso ficam em ondas separadas.
- O usuário que fez o merge de um merge request mergeado não é preservado.

11. 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 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.

12. 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 (importação e pós-importação), pela verificação e contagem no destino, pela reatribuição de usuários, pelo rollback, pela reescrita de referências (merge request por projeto) e pelo envio de imagens ao registry de destino.	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; com criação de usuários, reprova. A conta dona do token recebe as contribuições sem usuário correspondente.	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
Token de acesso da instância de origem	Usado pelo GitLab Congregate (listagem e Direct Transfer ou exportação), pelo preflight, pela contagem independente, pelo login no registry de origem (cópia de imagens), pela correspondência de usuários por e-mail, pela reaplicação de aprovadores e pelo arquivamento e desarquivamento da origem no corte.	Escopo <code>api</code> (inclui container registry e package registry, conforme a documentação GitLab). Direct Transfer: papel Owner nos grupos migrados ou administrador; o preflight reprova sem Owner no grupo de topo da origem. Método por arquivo, criação de usuários e correspondência por e-mail: administrador da origem.	Personal access token da instância de origem (em GitLab.com: token de acesso de grupo com papel Owner e escopo <code>api</code>), 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

13. 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
Tipo de origem	GitLab self-managed	Sim	GitLab.com ou GitLab self-managed.
URL da instância de origem	<code>https://gitlab.antigo.exemplo.gov.br</code>	Sim	Endereço HTTPS; <code>http://</code> gera alerta.
Versão e edição da origem	19.2, Enterprise Edition	Sim	Confere a regra de versões do método.
Nível de assinatura da origem	Premium	Sim	Free, Premium ou Ultimate. Com o nível errado, itens pagos são ignorados sem erro.
Grupo de topo (escopo) na origem	ti	Condicional: origem GitLab.com	Define o escopo da listagem; obrigatório na GitLab.com.
Endereço do container registry de origem	<code>registry.gitlab.antigo.exemplo.gov.br</code>	Condicional: cópia de imagens	Host e porta, sem protocolo.
Método	Direct Transfer	Não	Direct Transfer (padrão) ou por arquivo.

Destino

Informação	Exemplo	Obrigatório	Observação
URL da instância de destino	<code>https://gitlab.exemplo.gov.br</code>	Sim	
Versão e nível de assinatura do destino	19.4.1, Ultimate	Sim	
Grupo pai no destino	migrados	Sim	Grupo privado ou interno existente.

Informação	Exemplo	Obrigatório	Observação
Endereço do container registry de destino	registry.gitlab.exemplo.gov.br	Condicional: cópia de imagens	Host e porta, sem protocolo.
Reatribuição sem confirmação habilitada	Sim	Condicional: reatribuição em lote	Decisão do administrador do destino.

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: http://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.
Diretório da stack no host	/opt/pointer-ps/migracao	Não	Padrão /opt/pointer-ps/migracao
Acesso SSH para acompanhamento	usuario@host-migracao.exemplo.gov.br	Não	Túnel para a interface do GitLab Congregate.
Confirmação de conectividade destino → origem por HTTPS	Sim	Condicional: Direct Transfer	

Usuários

Informação	Exemplo	Obrigatório	Observação
Política de usuários	Usuários provisionados por LDAP; reatribuição em lote	Sim	Criar pelo GitLab Congregate, usar usuários já provisionados ou manter em usuários placeholder.
Campo de correspondência origem → destino	E-mail	Não	E-mail (padrão) ou username.
Planilha de correspondência de usuários	usuarios-mapa.csv (colunas origem, destino)	Não	Username na origem → username no destino; vale antes do campo de correspondência.
Incluir usuários inativos	Não	Condicional: criação de usuários	
Senha dos usuários criados	Senha aleatória	Condicional: criação de usuários	Senha aleatória (padrão) ou redefinição de senha pelo usuário.
Responsável pelos placeholders sem correspondente e prazo	Owner do grupo de topo; 10 dias	Condicional: Direct Transfer	

Ondas

Informação	Exemplo	Obrigatório	Observação
Planilha de ondas	ondas.csv (colunas onda, grupo, projeto)	Sim	Grupo inteiro (com subgrupos e projetos) ou projeto avulso, pelo caminho na origem; o mesmo item não se repete entre ondas. O inventário da origem serve de modelo.
Nome, data e descrição de cada onda	piloto; 10/11/2026; projetos 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 na origem durante a janela	Sim	Sim	

Pós-migração e corte

Informação	Exemplo	Obrigatório	Observação
Cópia das imagens de contêiner	Sim	Sim	Copiar pelo pipeline ou reconstruir no destino.
Reescrita de referências à origem	Relatório e merge request por projeto	Sim	Só relatório (padrão) ou merge request por projeto; revisor dos merge requests.
Arquivamento da origem no corte	Sim, 5 dias após cada onda	Sim	Sim ou não, e momento.
Itens de pós-importação a não recriar	Ambientes	Não	Por padrão todos são recriados.
Responsáveis pelos itens de tratamento manual	Runners e integrações: 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 da instância de origem	Nome, e-mail	Condicional: origem self-managed	
Administrador da instância de destino	Nome, e-mail	Sim	
Infraestrutura, rede e segurança	Nome, e-mail	Sim	Host, firewall, proxy e certificados.

14. Matriz de itens migrados

Nativo Migrado pelo método oficial (Direct Transfer ou, no método alternativo, importação por arquivo), disparado pelo pipeline

Pipeline Pointer Tratado pelo pipeline Pointer após a importação (pós-importação do GitLab Congregate ou cópia de imagens pelo host de migração)

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.

Grupos

Item	Tratamento	Observação
Subgrupos	Nativo	Subgrupos a menos no destino reprovam a onda na contagem independente.
Configurações do namespace	Nativo	
Badges de grupo	Nativo	A documentação GitLab lista badges como migrados; o runbook do GitLab Congregate orienta recriá-los. Adotado: migra, com conferência na onda piloto.
Uploads do grupo	Nativo	
Wiki de grupo	Nativo	Premium ou superior.
Membros do grupo	Nativo	Chegam em usuários placeholder e são reatribuídos em lote quando a reatribuição está no plano; convites pendentes não migram.
Grupo compartilhado com outros grupos	Pipeline Pointer	Recriado pelo pós-importação do GitLab Congregate. O Direct Transfer mapeia memberships compartilhadas como diretas; o compartilhamento só é preservado se a estrutura de grupos for migrada antes.
Webhooks de grupo	Pipeline Pointer	Recriados pelo pós-importação só com origem Premium ou superior; o token secreto não sai pela API e é redefinido no destino.
Deploy tokens de grupo	Ajuste manual	Excluídos do Direct Transfer.
Convites de membros pendentes	Não migra	Não suportado pelo Direct Transfer.
Eventos de auditoria (grupo e projeto)	Não migra	Não migrados pela exportação por arquivo, pelo Direct Transfer nem pelo GitLab Congregate.

Issues, épicos e planejamento

Item	Tratamento	Observação
Épicos, com boards e listas de épicos	Nativo	Premium ou superior.
Labels de grupo	Nativo	Prioridades de labels de grupo não são mantidas na importação (documentação GitLab).
Milestones de grupo e de release	Nativo	Título igual a milestone já existente no destino recebe sufixo único.
Boards e listas de grupo	Nativo	
Iterações e cadências de iteração	Nativo	As configurações das cadências não são migradas (ver notas).
Issues (comentários, eventos de estado, milestone e iteração, time tracking)	Nativo	Issues a menos no destino reprovam a onda.
Labels e milestones do projeto	Nativo	
Boards de issues	Nativo	
Designs	Nativo	
Issues vinculadas (linked items)	Não migra	A documentação GitLab lista como não suportado no Direct Transfer; a matriz do GitLab Congregate lista como suportado. Adotada a documentação GitLab, com conferência na onda piloto.

Repositório

Item	Tratamento	Observação
Repositório Git (commits, branches, tags)	Nativo	Arquivos com nome acima de 255 caracteres não migram. Branches e tags a menos no destino reprovam a onda.
Branches e tags protegidas	Nativo	Branches importadas seguem a proteção padrão de branch do grupo de destino: branch desprotegida na origem pode chegar protegida.
Branch padrão	Nativo	Migrada pelo Direct Transfer e reaplicada pelo pós-importação do GitLab Congregate.
Push rules do projeto	Nativo	Premium ou superior; também reaplicadas pelo pós-importação. Se a origem tem push rules de grupo, nenhuma push rule de projeto é importada (documentação GitLab).
Comentários de commit	Nativo	
Objetos LFS	Nativo	Não são baixados se o <code>.lfsconfig</code> apontar <code>lfs.url</code> para outro host. As estatísticas do destino podem atrasar.
Snippets do projeto	Nativo	Exige snippets habilitados no projeto de origem.
Wiki do projeto	Nativo	
Comentários de wiki	Não migra	Não suportado pelo Direct Transfer.
Releases e evidências de release	Nativo	
Espelhos remotos (push)	Ajuste manual	Não suportado pelo Direct Transfer.
Espelhamento pull	Ajuste manual	Projetos com espelho pull são apontados como ação manual pelo Levantamento da Plataforma.

Merge requests

Item	Tratamento	Observação
Merge requests (comentários, revisores, aprovadores, responsáveis, eventos, time tracking)	Nativo	Merge requests sem diff ou sem informação de origem não migram. Merge requests a menos no destino reprovam a onda.
Regras de aprovação de merge request	Pipeline Pointer	Recriadas pelo pós-importação com origem Premium ou superior. A instância de destino descarta o aprovador que ainda não é membro; o pipeline reaplica os aprovadores depois da redistribuição. Sem usuários correspondentes no destino, as regras ficam sem aprovadores.
Dependências entre merge requests	Não migra	Não suportado pelo Direct Transfer.

CI/CD

Item	Tratamento	Observação
Variáveis de CI/CD do projeto	Pipeline Pointer	Recriadas pelo pós-importação do GitLab Congregate.
Variáveis de CI/CD de grupo	Pipeline Pointer	Excluídas do Direct Transfer por poderem conter informação sensível; recriadas pelo pós-importação.
Histórico de pipelines (status, estágio, horários)	Nativo	Pipelines importados de merge requests mantêm o status da origem e podem satisfazer <i>Pipelines must succeed</i> , executar novo pipeline antes do merge de merge request importado.
Pipelines filhas (child pipelines)	Não migra	Não suportado pelo Direct Transfer.
Logs e artefatos de jobs	Não migra	Excluídos do Direct Transfer.
Agendamentos de pipeline	Nativo	As variáveis dos agendamentos são excluídas do Direct Transfer e recriadas pelo pós-importação.
Ambientes, inclusive ambientes protegidos	Pipeline Pointer	Recriados pelo pós-importação; ambientes protegidos com Premium ou superior.
Feature flags e listas de usuários de feature flags	Pipeline Pointer	Recriadas pelo pós-importação.
Deploy keys	Pipeline Pointer	Recriadas pelo pós-importação.
Allowlist do CI/CD job token	Pipeline Pointer	Recriada pelo GitLab Congregate só para grupos e projetos da onda que puderem ser resolvidos no destino, com permissões padrão.
Terraform states	Pipeline Pointer	Recriados pelo pós-importação.
Tokens de trigger de pipeline	Ajuste manual	Excluídos do Direct Transfer.
Deploy tokens do projeto	Ajuste manual	Excluídos do Direct Transfer.
Runners (de projeto, de grupo e de instância)	Ajuste manual	Registrados no destino na fase de pós-migração.
Agentes para Kubernetes	Ajuste manual	Não suportado pelo Direct Transfer.

Pacotes e registries

Item	Tratamento	Observação
Pacotes (Package Registry)	Pipeline Pointer	Recriados pelo pós-importação; a matriz do GitLab Congregate lista Maven, PyPI, npm, Generic, Helm, Composer e NuGet.
Imagens de contêiner (Container Registry)	Pipeline Pointer	Copiadas tag a tag pelo pipeline com o Docker do host de migração, quando a cópia de imagens está no plano; o rollback remove antes as imagens.

Configurações, membros e integrações do projeto

Item	Tratamento	Observação
Configurações do projeto (propriedades, recursos, Auto DevOps, avatar, badges, política de expiração do registry, Service Desk)	Nativo	
Membros do projeto	Nativo	Chegam em usuários placeholder e são reatribuídos em lote quando a reatribuição está no plano.
Projeto compartilhado com grupos	Pipeline Pointer	Recriado pelo pós-importação do GitLab Congregate.
Webhooks do projeto	Pipeline Pointer	Recriados pelo pós-importação com origem Premium ou superior; o token secreto não é exposto pela API e é redefinido no destino.
Uploads (anexos)	Nativo	
Integrações de projeto (por exemplo Jira e Slack)	Ajuste manual	Não suportadas pelo GitLab Congregate; reconfiguradas no destino na fase de pós-migração.
Relatórios de vulnerabilidade	Nativo	Migrados desde a versão 17.7, sem o status (documentação GitLab).

Instância

Item	Tratamento	Observação
System hooks	Ajuste manual	O GitLab Congregate só migra system hooks com uma opção que o pipeline não habilita.

Notas da matriz

- **A confirmar na onda piloto** (tratamento não classificado pelo pipeline Pointer; definido no plano de ondas e conferido na onda piloto): configurações das cadências de iteração, push rules de grupo e campos personalizados de grupo e projeto (a documentação GitLab lista os três como não suportados pelo Direct Transfer); domínios do GitLab Pages (não suportados pelo Direct Transfer); tokens de acesso pessoais, de projeto e de grupo e tokens criptografados (a documentação GitLab informa que usuários e os tokens que criam são excluídos da migração); configurações da instância, aparência, broadcast messages e variáveis de instância (fora do Direct Transfer; o GitLab Congregate informa que ainda não suporta).
- **A confirmar na onda piloto** (sem tratamento definido nas fontes): módulos Terraform do Infrastructure Registry (a documentação GitLab lista como não suportado pelo Direct Transfer e a matriz do GitLab Congregate como suportado, sem etapa específica no pipeline); relação de fork (suportada segundo a matriz do GitLab Congregate, sem etapa específica no pipeline); chaves SSH e GPG dos usuários criados pelo GitLab Congregate (a documentação do GitLab Congregate orienta verificar as chaves no destino e criar novas se faltarem).
- Nível de assinatura: com o nível da origem informado errado, o GitLab Congregate ignora sem erro itens pagos como regras de aprovação, webhooks de grupo e de projeto e push rules.
- O pós-importação do GitLab Congregate desarchive temporariamente os projetos arquivados na origem e no destino para aplicar os itens e depois os arquiva de novo.
- URLs em notas e descrições só são reescritas para links do próprio grupo ou projeto migrado; links para outros projetos da origem permanecem (a reescrita de referências nos arquivos dos repositórios é tratada à parte).
- Após o início da migração de uma onda, alterações na origem podem não ser copiadas para o destino.
- Método por arquivo: as diferenças em relação a esta matriz estão na seção "Método por arquivo: diferenças em relação ao Direct Transfer".

15. 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.	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
☐	Direct Transfer habilitado no destino.	Preflight automático
☐	Grupo pai no destino existente e privado ou interno.	Preflight automático
☐	Origem acessível; versões dentro da regra do Direct Transfer.	Preflight automático
☐	Grupo de topo da origem encontrado e token da origem com papel Owner nele ou administrador.	Preflight automático
☐	Reatribuição sem confirmação habilitada no destino, quando a reatribuição em lote está no plano (alerta).	Preflight automático
☐	Configuração do GitLab Congregate validada.	Preflight automático
☐	Direct Transfer habilitado também na origem self-managed.	Reunião de prontidão
☐	Instância de destino alcança a origem por HTTPS.	Reunião de prontidão
☐	Sidekiq do destino configurado para importações; espaço em <code>/tmp</code> na origem e no destino; <code>proxy_download</code> ou acesso ao object storage da origem.	Reunião de prontidão
☐	<i>Default minimum role required to create projects</i> diferente de <i>No one</i> e snippets habilitados na origem.	Reunião de prontidão
☐	Método por arquivo: versões e fonte de importação do destino conferidas; tokens de administrador na origem e no destino.	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
☐	Política de usuários definida; no método por arquivo, usuários criados ou provisionados antes da primeira onda.	Reunião de prontidão

☐	Item	Verificado por
☐	Host de migração provisionado conforme a especificação, com saída HTTPS para origem, destino, gitlab.com, registry.gitlab.com e Docker Hub.	Cliente
☐	Tokens da origem e do destino emitidos com escopos, papéis e validade exigidos e cadastrados como variáveis protegidas.	Cliente
☐	Congelamento de escrita comunicado aos usuários da origem para a janela de cada onda.	Cliente
☐	Com cópia de imagens: Docker do host confiando na CA do registry e da instância de destino, e disco para a maior imagem.	Cliente

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

PR-MIG-01 · versão 1.2 · 1 de outubro de 2026. GITLAB is a trademark of GitLab Inc. in the United States and other countries and regions. A Pointer é parceira GitLab Select.