

PULSE / Civil 3D

Guida e comandi

Autodesk Civil 3D 2023-2027

Installazione, primi passi ed esempi in linguaggio naturale. I nomi tecnici dei comandi restano invariati.

Fonte verificata: <https://github.com/LuDattilo/pulse-for-civil3d>

Branch: main | Commit: 1da6e3cd6a5bd8b2ff4652e0aa7ca88ab0777e8f

2026-10-04 | IT

Descrizioni in italiano. Nomi dei comandi, parametri e valori tecnici restano invariati. Catalogo delle sorgenti del 4 ottobre 2026; traduzioni aggiornate l'8 ottobre 2026.

<https://pulse-ai.it/guida-civil3d.html>

Installazione e collegamento

Chiudi Civil 3D e installa il pacchetto EXE dedicato. Riavvia il programma, apri un disegno e accendi Pulse dalla scheda dedicata o con PULSECONNECT. Verifica che il client AI veda pulse-civil3d. I sorgenti prevedono 2023-2027; le prove runtime documentate nel README riguardano Civil 3D 2025.

Collega un client AI compatibile con MCP: Claude, Codex, opencode o un altro client, anche con modelli locali.

Apri il progetto e descrivi l'obiettivo. Verifica il risultato e approva le modifiche quando richiesto.

Lavora su una copia del progetto. Chiedi una lettura o un'anteprima prima delle modifiche, controlla la selezione e approva solo le operazioni attese.

Alla prima registrazione idonea, Trial Premium personale di 14 giorni. Una nuova richiesta non rinnova una trial già attivata.

Richieste di esempio

- > Confronta queste due superfici e mostrami i volumi di scavo e riporto.
- > Elenca i tracciati e leggi il profilo selezionato.
- > Trova i punti del gruppo rilievo ed esportali nel formato richiesto.

FUNZIONALITÀ

Superfici e volumi

Leggi quote, aree e volumi; crea superfici da punti e superfici di confronto, con anteprima dove prevista.

Tracciati e profili

Consulta geometrie, progressive, offset, profili e sezioni; crea tracciati da polilinee e profili di superficie.

Punti e rilievo

Interroga gruppi e punti COGO, importa ed esporta punti e gestisci le proprietà definite dall'utente.

Corridoi e quantità

Elenca e ricostruisci corridoi; consulta sezioni, liste materiali e quantità disponibili.

Reti e disegno

Leggi reti, condotte e pozzetti, verifica la rete e consulta oggetti, layer e stili del disegno.

Dynamo e automazioni

Consulta lo stato di Dynamo, genera ed esegui grafi; il codice personalizzato resta soggetto alle impostazioni di sicurezza.

Catalogo dei comandi

Descrizioni in italiano. Nomi dei comandi, parametri e valori tecnici restano invariati. Catalogo delle sorgenti del 4 ottobre 2026; traduzioni aggiornate l'8 ottobre 2026.

Tracciati

`get_alignment_geometry`

Restituisce la geometria di un tracciato, elencando le entità nell'ordine: tipo (Line, Arc, Spiral, SpiralCurveSpiral, ...) e, per ogni sottoentità, tipo, progressiva iniziale e finale, lunghezza e, per gli archi, raggio. Le entità sono paginate tramite limit e offset. disconnectedSkipped segnala entità disconnesse che non sono state elencate. Progressive e lunghezze sono espresse nelle unità del disegno.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs

`get_feature_line`

Restituisce una linea caratteristica, individuata per nome o handle, come indicato da list_feature_lines: nome, handle, sito, stile, numero di punti, lunghezze 2D e 3D, quote e pendenze minime e massime. Per i punti restituisce indice, x, y, z, bulge del segmento iniziante nel punto e pendenze in ingresso e uscita, quando presenti: il primo punto non ha pendenza in ingresso, l'ultimo non ha pendenza in uscita. I punti sono paginati tramite limit e offset; le coordinate sono nelle unità del disegno (WCS: X verso est, Y verso nord).

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs

`get_material_quantities`

I volumi di taglio, riempimento e utilizzabili di un elenco di materiali lungo le linee di campionamento di un group: una voce per linea di campionamento con i volumi incrementali e cumulativi nella relativa stazione, quindi il totale (il cumulativo nell'ultima linea di campionamento). Assegnare un nome all'allineamento e alla lista dei materiali come li restituisce list_material_lists; 'group' seleziona il gruppo di linee di sezione quando l'allineamento ne ha più di uno. Le stazioni sono impaginate con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs

`get_point_at_station`

Key di ciascun elemento [stazione, offset] su un allineamento, fino a 5000 per chiamata. Una stazione fuori portata dell'allineamento non è un errore: ritorna con xey null e fuori true, e la risposta conta gli elementi dentro e fuori. Stazione e offset utilizzano lo stazionamento dell'allineamento, in unità di disegno.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs

`get_profile_data`

I dati del profilo di un alignment: i PVI (stazione, elevazione e pendenza in entrata e in uscita laddove esistono: il primo PVI non ha pendenza in entrata, l'ultimo nessuna pendenza in uscita) e, quando vengono fornite le stazioni, l'elevazione su ciascuna di esse. I voti sono rapporti ($0.02 = 2\%$). Le progressive utilizzano lo stazionamento dell'allineamento, in unità di disegno. Denominare il profilo come `list_alignments` lo restituisce nei profili di allineamento.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs`

`get_section_data`

Una sezione con i suoi punti geometrici, per maniglia o per allineamento e stazione (le due modalità si escludono a vicenda). La sezione fornisce la stazione, l'origine (`sourceName` e `sourceType`: `TinSurface`, `Classificazione`, `Corridoio`, `CorridorSurface`, `PipeNetwork`, `Materiale`, `GridSurface`), elevazione minima e massima, sinistra e destra offset e la larghezza dell'andana dove il tipo di sezione li supporta, nonché il relativo stile. I punti sono impaginati con `limit` e `offset` e forniscono, per punto, l'indice, offset e l'elevazione nel piano di sezione e `segmentTo` - l'indice del punto a cui va il suo segmento, -1 quando il punto non inizia alcun segmento. Con l'allineamento e la stazione lo strumento seleziona la linea campione più vicina entro la tolleranza (predefinito 0.000001); quando non ce n'è, la risposta è 100 con stazioni valide fino a 50. Quando la linea campione prevede più sezioni, viene descritta la prima e `otherSections` elenca le altre per handle.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs`

`get_station_offset`

La stazione e offset su un allineamento di ciascun punto `[x, y]` in unità di disegno (WCS: X est, Y nord), fino a 5000 per chiamata. Un punto fuori dal campo dell'allineamento non è un error: torna con stazione e offset null e fuori true, e la risposta conta i punti dentro e fuori. Stazione e offset utilizzano lo stazionamento dell'allineamento, in unità di disegno.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs`

`list_alignments`

Elenco degli allineamenti del disegno attivo, ordinati per nome: nome, maniglia, tipo (Asse centrale, Offset, CurbReturn, Utilità, Binario), sito, lunghezza, stazione iniziale e finale, stile e nomi e maniglie dei profili. Le stazioni, le lunghezze e gli offset sono espressi in unità di disegno (vedi `get_drawing_info`). Impaginato con `limit` e `offset`.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs`

`list_feature_lines`

Elenco delle linee caratteristiche dei siti del disegno attivo, ordinate per sito e nome: nome, maniglia, sito, stile, numero di punti, lunghezza 2D e 3D, elevazione minima e massima e pendenza minima e massima. Un sito (nome o handle) restringe l'elenco alle linee caratteristiche di quel sito; le linee caratteristiche che non si trovano in nessun sito non vengono elencate perché Civil 3D API raggiunge una linea caratteristica solo attraverso il relativo sito. Lunghezze e quote sono espresse in unità di disegno (vedi `get_drawing_info`); i voti sono rapporti ($0.02 = 2\%$). Impaginato con `limit` e `offset`.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs`

`list_material_lists`

Elenca gli elenchi dei materiali dei gruppi di linee di sezione del disegno, ordinati per allineamento e name: ciascun gruppo con i suoi materiali e il relativo tipo di quantità (Scarico, Riempimento, CutFill, Movimento terra, Struttura). Questi sono i nomi e le guide che `get_material_quantities` prende. Passa 'alignment' per restringere la risposta. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs

`list_sample_lines`

Elenca le linee di sezione del disegno attivo, ordinate per tracciato, gruppo e station: il tracciato e il gruppo di appartenenza (nome e maniglia), il nome della linea di sezione, maniglia, numero, stazione, lockToStation, il numero di sezioni ad essa collegate (sectionCount) e il suo stile. Un allineamento (nome o handle) restringe l'elenco a quell'allineamento. Una linea di sezione appartiene a un gruppo e un gruppo a un tracciato, pertanto non viene elencata alcuna linea di sezione di un disegno senza tracciati. Le stazioni sono in unità di disegno (vedi `get_drawing_info`). Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\AlignmentTools.cs

Operazioni di base

`find_objects`

Trova gli oggetti dello spazio modello del disegno attivo, filtrati per tipologia, layer e nome. tipo e layer accettano i caratteri jolly * and ?; il tipo corrisponde al nome della classe .NET (Polilinea, TinSurface) o al nome DXF (LWPOLYLINE); il nome corrisponde esattamente, ignorando le maiuscole e minuscole. Risposte handle, tipo, dxfName, livello e nome, ordinati per handle. Impaginato con limit e offset. Passare le maniglie a `get_object_properties` o `select_objects`.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\BaseTools.cs

`get_coordinate_settings`

Il sistema di coordinate assegnato al disegno attivo (codice, descrizione, proiezione, datum, unità, categoria; assegnato false quando non ce n'è, la voce "." della libreria conta come nessuno) e la trasformazione da locale a griglia parameters: punti di riferimento e di rotazione, la rotazione verso il nord o l'azimut della griglia in gradi, il fattore di scala della griglia quando l'utente lo definisce e il fattore di scala del livello del mare. 'transformation' è {enabled: false} mentre le impostazioni di trasformazione sono disattivate. Confrontare con Impostazioni Carta > Trasformazione di Civil 3D.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\BaseTools.cs

`list_layers`

Elenca i layer del disegno attivo, ordinati per name: nome, maniglia, disattivato, congelato, bloccato, se si tratta del layer corrente, il nome visualizzato del colore (quello mostrato dal gestore layer, quindi localizzato) e il nome del tipo di linea. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\BaseTools.cs

list_styles

Elenca gli stili del disegno attivo. Insieme al tipo (superficie, allineamento, profilo, profileView, tubazione, struttura, punto, alignmentLabelSet, profileLabelSet) elenca gli stili di quel tipo, ordinati per name: nome e maniglia. Senza tipo risponde quanti stili di ogni tipo contiene il disegno. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\BaseTools.cs

select_objects

Selezionare gli oggetti denominati in base alle relative maniglie in Civil 3D (la selezione pickfirst), nell'ordine indicato e, per impostazione predefinita, inquadrarli nella vista attiva. Il disegno non cambia. Le maniglie non presenti nel disegno ritornano in notFound e non vengono selezionate; una selezione vuota cancella quella corrente. Leggi le maniglie da find_objects, list_alignments, list_corridors o dagli altri strumenti di elenco.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\BaseTools.cs

Codice

analyze_send_code_gaps

Quali operazioni continuano a richiedere send_code_to_civil3d: le sue chiamate nel registro di controllo (~/.pulse/civil3d/logs/audit.jsonl) raggruppate per `purpose`, le più frequenti per prime, ciascuna con il numero di chiamate, di esecuzioni reali e la prima e l'ultima data. Gli scopi sono raggruppati per testo esatto (tagliato, senza distinzione tra maiuscole e minuscole). Il codice stesso non è mai nel registro. Risposta da server: funziona senza Civil 3D. Usalo per vedere quale strumento dedicato potrebbe essere più utile.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\CodeTools.cs

send_code_to_civil3d

LAST RESORT: esegue il codice C# in Civil 3D. Utilizzarlo solo quando nessuno strumento dedicato copre l'operazione (vedi tools/list). `code` è il corpo di un metodo che restituisce oggetto. Riceve documento (Document), database (Database), editor (Editor), civilDocument (CivilDocument) e transazione (Transaction), già open: apri oggetti con esso e non eseguirne mai il commit, l'interruzione o l'eliminazione. Non avviare mai una transazione con il tuo own: StartTransaction, StartOpenCloseTransaction e TopTransaction verranno rifiutati. Pulse esegue il commit quando il codice ritorna e ripristina tutto quando genera un'eccezione. Il valore restituito ritorna come JSON (numeri, stringhe, liste, oggetti anonimi); un oggetto AutoCAD ritorna come handle, tipo, livello e nome. Civil 3D 2025-2027 compila C# 13 (Roslyn); 2023-2024 compila C# solo 5 (CodeDom): nessuna stringa \$""", ?., nameof, out var, pattern `is`, tuple o funzioni locali. Nessun file, rete, processo, registro, riflessione o dinamica. I comandi in coda (SendStringToExecute, Editor.Command) e le operazioni sui file DWG/DXF vengono rifiutati. `purpose` è required: una riga su cosa fa il codice e perché nessuno strumento dedicato lo copre; il registro di controllo lo mantiene chiaro, mai il codice. L'anteprima (dryRun omessa) non compila e non esegue nulla. L'esecuzione (dryRun:false e l'anteprima `drawing`) mostra all'utente lo scopo e l'intero codice in Civil 3D e viene eseguita solo dopo un sì, quindi non è necessario un consenso separato nella chat. Rifiutato con PermissionDenied mentre EnableCodeExecution è disattivato in ~/.pulse/civil3d/settings.json: quindi non riprovare. Una ANNULLA (UNDO) in Civil 3D annulla una sequenza.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\CodeTools.cs

Modellatori

`list_corridors`

Elenca i modellatori della Carta attiva, ordinati per name: nome, handle, se non aggiornato, le linee di base (tracciato, profilo e nome di ciascuna regione, stazione iniziale e finale e assemblaggio) e i nomi delle superfici del modellatore. Le stazioni sono in unità di disegno. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\CorridorTools.cs

Disegno

`get_drawing_info`

Descrivere il drawing attivo: unità lineari e angolari, scala del disegno, sistema di coordinate (codice, descrizione, proiezione, datum), quanti oggetti Civil 3D di ogni tipo contiene (superfici, tracciati, reti di condotte, corridoi, punti COGO, gruppi di punti, siti, linee caratteristiche senza sito, intersezioni) e quali superfici, corridoi e gruppi di punti non sono aggiornati. Le lunghezze e le altezze in ogni altra risposta utilizzano queste unità di disegno.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DrawingTools.cs

`get_object_properties`

Leggere le proprietà pubbliche degli oggetti tramite handle. Numeri, testi, booleani ed enumerazioni (come nomi) vengono forniti così come sono, punti come {x,y,z}, riferimenti a oggetti come {handle, name} (StyleId fornisce l'handle e il nome dello stile). Le raccolte e altre proprietà complesse vengono ignorate e conteggiate, le proprietà che generano eccezioni vengono elencate con il tipo di eccezione, i valori null vengono elencati per nome. Fino a 100 maniglie e proprietà 200 per oggetto; le maniglie non presenti nel disegno ritornano in notFound.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DrawingTools.cs

`get_selected_objects`

Elenca gli oggetti selezionati dall'utente in Civil 3D prima della chiamata (selezione pickfirst), nell'ordine di selezione: handle, tipo .NET, nome DXF, layer e nome. Paginazione tramite limit e offset; il risultato include totalCount, returnedCount, truncated e nextOffset. Senza selezione l'elenco è vuoto e selectionStatus ne indica il motivo. Passare gli handle a get_object_properties.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DrawingTools.cs

Dynamo

dynamo_generate_graph

Scrivere un grafico Dynamo completo (.dyn) per questo Civil 3D da un grafico description: nodi (input di valori, dispositivi di scorrimento, blocco codice, orologio, funzioni di libreria di FunctionSignature, script Python, nodi del pacchetto), connettori, gruppi e note, nel formato delle competenze dynamo-python. I nodi della libreria devono essere nodi reali di questo Civil 3D: FunctionSignature è Namespace.Type.Member@type1,type2 e uno sconosciuto viene rifiutato con quelli più vicini. Vengono controllati i nodi Python (linea motore, sandbox, idiomi Python 2). Il grafico viene convalidato prima di essere salvato; non corre niente. Prima l'anteprima (dryRun o messo), poi dryRun: false con il disegno restituito. DesignScript pitfalls: 'cond ? x : []' restituisce sempre []; utilizzare '[[, x][cond ? 1 : 0]'; DSCore.Math.Abs su un elenco vuoto fallisce con un sovraccarico ambiguo (Abs(double) o Abs(int)); PullOntoPlane fornisce una curva generica.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DynamoTools.cs

dynamo_get_status

Lo stato di Dynamo in Civil 3D connesso (non caricato, caricato, avviato senza o con la relativa finestra), la relativa versione, il percorso di esecuzione che verrà utilizzato da dynamo_run_graph e il risultato dell'ultima esecuzione del grafico. Sola lettura.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DynamoTools.cs

dynamo_list_graph_io

Leggere un grafico Dynamo (.dyn) dal disco ed elencare cosa può impostare dynamo_run_graph e read: i suoi input (id, nome, tipo, valore corrente), i suoi output, la versione di Dynamo con cui è stato salvato, i suoi nodi e motore Python. Risposta da Pulse server: funziona senza Civil 3D. Chiamalo prima di dynamo_run_graph e passa solo i nomi di input elencati in inputValues.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DynamoTools.cs

dynamo_run_graph

Esegui un grafico Dynamo (.dyn) in Civil 3D connesso, con nuovi valori per i suoi input e leggi i suoi output e gli avvisi sui nodi. Chiama dynamo_list_graph_io first: inputValues accetta solo i nomi di input elencati. Il grafico può modificare il disegno, quindi questo è un write: prima anteprima (dryRun o messo), poi dryRun: false con il disegno che restituisce; l'utente conferma ogni esecuzione. Rifiutato mentre la finestra di Dynamo è aperta. La route A esegue Dynamo senza la relativa finestra e restituisce output; il percorso B (AECCRUNDYNAMOSCRIPT) non restituisce output. Il percorso B non può leggere il nodo errors: riuscito significa che il comando è stato eseguito. Entrambi i percorsi restituiscono databaseEvents, gli oggetti aggiunti, modificati e cancellati nel disegno. Fino a 300 s.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\DynamoTools.cs

Informazioni del server

get_session_info

Quali pezzi Pulse sono in esecuzione e come sono impostati up: il server (versione, protocollo, moduli, impostazioni e file di registro), le istanze Civil 3D con Pulse acceso e quello in uso e, quando uno è connesso, i suoi disegni aperti, la modalità di sola lettura e lo stato di Dynamo. Risposte anche senza Civil 3D: chiamalo PRIMA quando manca uno strumento o una chiamata fallisce con uno BridgeFailure.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\MetaTools.cs

say_hello

Testare la connessione a Civil 3D. Risposte a cui ha risposto Civil 3D (prodotto, anno, versione AutoCAD), la sua lingua (LOCALE), il disegno attivo e le versioni del plugin e del protocollo. Non necessita di disegno aperto.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\MetaTools.cs

Reti di condotte

add_structure

Aggiungere una struttura a una rete di condotte esistente. La famiglia e la dimensione provengono dal dominio della struttura dell'elenco delle parti, poiché list_parts_lists le restituisce; 'x' e 'y' sono la posizione in pianta, 'z' l'elevazione della struttura (ometterla per posizionarla sulla superficie di riferimento della rete), 'rotation' è in gradi e 'applyRules' per impostazione predefinita è true come fa Aggiungi struttura di Civil 3D. L'omissione di dryRun è un'anteprima. Con dryRun false la struttura viene aggiunta in una transazione di scrittura.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PipeTools.cs

check_pipe_network

I problemi di una rete, una voce per tubazione o struttura con il nome dell'assegno. Sempre: missing_structure_start/end (un tubo senza struttura all'estremità), no_pipes (una struttura senza tubi), zero_length, adverse_slope (il tubo sale lungo la sua direzione). Con i limiti given: slope_below_minimum e slope_above_maximum (percentuale), cover_below_minimum (unità di disegno, dalla superficie di riferimento; cover_unavailable quando non ce n'è).

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PipeTools.cs

get_pipe_network

I tubi e le strutture di una rete. Ogni pipe: maniglia, famiglia, dimensione, dritta o curva, le sue due estremità, quote invertite (la linea centrale meno metà dell'altezza interna), pendenza, lunghezze 2D e 3D, altezza interna e diametro, copre quando la rete ha una superficie di riferimento e le strutture alle sue estremità. Ogni structure: famiglia, dimensione, tipo, ubicazione, quote del bordo e del pozzetto, numero di tubi. I tubi sono impaginati con limit e offset; le strutture sono sempre tutte elencate.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PipeTools.cs

list_parts_lists

Elenca gli elenchi delle parti del disegno attivo, ordinati per nome, ciascuno con le famiglie di tubazioni e strutture e i relativi sizes: i nomi esatti presi dagli strumenti di creazione. Passa 'partsList' per un elenco. Impaginato con limit e offset; al massimo 500 taglie per famiglia (sizesTruncated quando di più).

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PipeTools.cs

`list_pipe_networks`

Elenca le reti di tubazioni del disegno attivo, ordinate per name: maniglia, elenco delle parti, superficie di riferimento e allineamento e numero di tubi e strutture. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PipeTools.cs

`list_structures`

Elencare le strutture delle reti di tubazioni del disegno, ordinate per rete e name: maniglia, famiglia, dimensione, tipo, posizione, rotazione in gradi, quote del bordo e del pozzo e profondità del pozzo. Passa 'network' per restringere la risposta a una rete. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PipeTools.cs

Punti

`get_cogo_point_udps`

I valori UDP dei punti COGO, raggruppati per classificazione, per maniglia punto (maniglie da 1 a 100). 'classification' si restringe a una classificazione. Un valore che non può essere letto viene tralasciato. I valori sono dati, non geometry: i numeri non vengono arrotondati.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PointTools.cs

`get_cogo_points`

I punti COGO di un gruppo di punti, di un intervallo di numeri di punti, o dell'intero drawing: numero, coordinate (locali, griglia o geografiche degrees: vedere 'coordinates'), nome, elevazione, descrizione grezza e completa, ordinati per numero. Utilizzare pointGroup O fromNumber/toNumber, non entrambi. Un'elevazione mai impostata ritorna come null. Le coordinate locali e della griglia sono espresse in unità di disegno (WCS: X est, Y nord). Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PointTools.cs

`list_point_groups`

Elenca i gruppi di punti del disegno attivo, ordinati per name: nome, handle, numero di punti e se il gruppo non è aggiornato (una ricostruzione è in sospeso). Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PointTools.cs

`list_udp_classifications`

Elenca le classificazioni punto UDP del disegno attivo, ordinate per name: ciascuna classificazione con i suoi UDP, ciascuna con nome, tipo (stringa, double, intero, booleano o enumerativo), descrizione, valore predefinito e, per le enumerazioni, i valori consentiti. Usalo per trovare i nomi e i tipi esatti di set_cogo_point_udps. Impaginato con limit e offset.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\PointTools.cs

Siti

`get_parcel`

Una particella, per handle o per sito e number: gli stessi campi di `list_parcel`s più le proprietà definite dall'utente (UDP) della particella raggruppate per classificazione, lette con Civil 3D API. A UDP viene tralasciato il pacco mai impostato, senza default. Civil 3D API non espone l'area o il perimetro della particella. Le coordinate sono in unità di disegno (vedi `get_drawing_info`).

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\SiteTools.cs`

`list_parcel`s

Elenca le particelle dei siti del disegno attivo, ordinate per sito e particella number: handle, numero, nome, sito, indirizzo, codice fiscale, centroide (x, y, z) e stile. Un sito (nome o handle) restringe l'elenco ai lotti di quel sito. Le coordinate sono in unità di disegno (vedi `get_drawing_info`). Civil 3D API non espone alcuna area della particella o perimetro: leggerli in Civil 3D dalle etichette dell'area della particella. Impaginato con `limit` e `offset`.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\SiteTools.cs`

Superfici

`add_surface_boundaries`

Aggiungere contorni a una superficie TIN dalle linee e polilinee denominate tramite le relative maniglie (da 1 a 500) e ricostruirla. 'type' è uno tra esterno (definisce l'estensione della superficie), `dataClip` (ritaglia i dati), mostra, nascondi; 'midOrdinate' è in unità di disegno (predefinito 1.0); Il valore predefinito di 'nonDestructiveBreakline' è false. Omettendo `dryRun` è un preview: nomina la superficie e gli oggetti. Con `dryRun` false i confini vengono aggiunti in una transazione di scrittura.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs`

`add_surface_breaklines`

Aggiungere linee di discontinuità standard a una superficie TIN dalle linee e polilinee denominate tramite le relative maniglie (da 1 a 500) e ricostruirla. 'midOrdinate' è la distanza dell'ordinata media in unità di disegno (default 1.0); gli opzionali 'maximumDistance', 'weedingDistance' e 'weedingAngle' hanno come impostazione predefinita 0, che aggiunge ogni vertice. Omettendo `dryRun` è un preview: nomina la superficie e gli oggetti. Con `dryRun` false le linee di discontinuità vengono aggiunte in un'unica transazione di scrittura.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs`

`add_surface_point_file`

Aggiungi un file di punti alla definizione di una superficie TIN e ricostruiscilo. Il formato deriva da 'format' (un nome che Civil 3D conosce) o dall'estensione del file. Omettendo `dryRun` si ottiene un preview: nomina la superficie, il formato e i punti del file. Con `dryRun` false il file viene aggiunto in una transazione di scrittura.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs`

`create_volume_surface`

Creare una superficie volumetrica TIN confrontando due superfici, per volumi di sterro e riporto (scavi e sterri). Omettendo `dryRun` è un preview: non cambia nulla e la risposta nomina lo stile che utilizzerebbe. Con `dryRun` false la superficie viene creata con lo stile scelto (il primo stile di superficie del disegno quando lo stile viene omesso) e ricostruita; la risposta nomina lo stile e, quando leggibile, i volumi di taglio, riempimento e netto non rettificati. Il nome deve essere inutilizzato e le due superfici devono essere diverse. I volumi sono espressi in unità di disegno cubiche.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs

`get_surface_elevations`

Leggere l'elevazione di una superficie nei punti indicati come [x, y] in unità di disegno (WCS: X est, Y nord), fino a 5000 per chiamata. Un punto esterno alla superficie non è un error: ritorna con z null e esterno true, e la risposta conta i punti interni ed esterni. Assegna un nome alla superficie per nome o handle, poiché `list_surfaces` li restituisce.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs

`get_surface_volumes`

Leggere i volumi di un volume surface: volumi di sterro, riporto e netti non rettificati e rettificati, i fattori di sterro e riempimento e le superfici di base e di confronto confrontate. I volumi sono espressi in unità di disegno cubiche. Assegnare un nome alla superficie per nome o handle, poiché `list_surfaces` le restituisce (`isVolume` true); una superficie che non è una superficie volumetrica viene rifiutata con l'elenco di cosa è.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs

`list_surfaces`

Elenco delle superfici del disegno attivo, ordinate per nome: nome, maniglia, tipologia (`TinSurface`, `GridSurface`, `TinVolumeSurface`, `GridVolumeSurface`), stile, quota minima, massima e media, numero di punti, e per le superfici TIN il numero di triangoli e le aree 2D e 3D (griglia surfaces: le aree). `outOfDate` significa che la superficie necessita di una ricostruzione; vuoto significa che non ha triangoli o celle. Impaginato con `limit` e `offset`. Prospetti e aree sono espressi in unità di disegno (vedi `get_drawing_info`).

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs

`paste_surface`

Incolla una superficie in un'altra superficie TIN e ricostruiscila, come fa Incolla superficie di Civil 3D. Assegna un nome alla superficie in cui incollare e alla superficie di origine da cui incollare; devono differire e il target deve essere una superficie TIN. Omettendo `dryRun` si ottiene un preview: nomina entrambe le superfici. Con `dryRun` false l'incollamento avviene in un'unica transazione di scrittura.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\SurfaceTools.cs

Operazioni di scrittura

`create_alignment_from_polyline`

Crea un allineamento da una Polilinea, Polilinea2d o Polilinea3d. Omettendo `dryRun` è un preview: non cambia nulla e la risposta nomina il livello, lo stile e il set di etichette che utilizzerebbe. Con `dryRun` false l'allineamento viene creato sul layer le impostazioni del disegno danno allineamenti (il layer corrente quando quel layer è mancante), con lo stile scelto e il set di etichette (il primo del disegno quando omesso), facoltativamente all'interno di un sito, facoltativamente aggiungendo curve tra tangenti e facoltativamente cancellando la polilinea. Il nome deve essere inutilizzato.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs`

`create_cogo_points`

Crea punti COGO nelle posizioni indicate, ciascuna con una descrizione opzionale. Omettendo `dryRun` si ottiene un preview: non cambia nulla e la risposta dice quanti punti verrebbero creati. Con `dryRun` false i numeri seguono le impostazioni dei punti del disegno e la risposta elenca i numeri e le maniglie create. Una chiamata crea fino a 5000 punti.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs`

`create_surface_from_points`

Crea una superficie TIN da un gruppo di punti e ricostruiscila. Omettendo `dryRun` è un preview: non cambia nulla e la risposta nomina lo stile che utilizzerebbe. Con `dryRun` false la superficie viene creata con lo stile scelto (il primo stile di superficie del disegno quando lo stile è omesso) e costruita dai punti del gruppo di punti. Il nome deve essere inutilizzato; la risposta nomina lo stile e il gruppo di punti utilizzati.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs`

`create_surface_profile`

Creare un profilo di superficie (EG) per un allineamento. Omettendo `dryRun` è un preview: non cambia nulla e la risposta nomina il livello, lo stile e il set di etichette che utilizzerebbe. Con `dryRun` false il profilo viene creato sul layer che le impostazioni del disegno danno profili (il layer corrente quando quel layer manca), con lo stile scelto e l'etichetta impostata (il primo del disegno quando omesso); il nome predefinito è '<allineamento> - <superficie>' e deve essere inutilizzato nell'allineamento. La risposta nomina il tipo, le stazioni e il layer, lo stile e il set di etichette utilizzati.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs`

`delete_objects`

Cancella gli oggetti denominati tramite le rispettive maniglie (da 1 a 500), poiché `find_objects` li restituisce. Omettendo `dryRun` è un preview: elenca il tipo, il nome e il livello di ogni oggetto e rifiuta gli oggetti Civil riporta come in uso da un altro oggetto, quindi un oggetto referenziato non cancella mai la metà di quanto richiesto. Con `dryRun` false ogni oggetto si cancella in una sola scrittura transaction: tutto o niente.

Ambito: | `sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs`

export_points

Scrivere i punti COGO in un file di punti su questo machine: l'intero disegno o un gruppo di punti con 'pointGroup'. Il formato deriva da 'format' (un nome che Civil 3D conosce) o dall'estensione del file. Omettendo dryRun è un preview: nomina il percorso, il formato e quanti punti andrebbero. Con dryRun false il file viene scritto dopo la conferma e la risposta dice quanti punti sono andati.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs

import_points

Importa punti COGO da un file di punti su questa macchina. Il formato deriva da 'format' (un nome che Civil 3D conosce) o dall'estensione del file; 'pointGroup' inserisce i punti importati in quel gruppo. Omettendo dryRun è un preview: nomina il formato e conta i punti del file. Con dryRun false i punti vengono importati in una sola transazione di scrittura e la risposta dice quanti ne sono arrivati.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs

rebuild_corridor

Ricostruire un corridoio dopo la modifica degli elementi da cui dipende (tracciati, profili, superfici). Omettendo dryRun si ottiene preview: indica se il corridoio non è aggiornato e quante regioni ha. La ricostruzione può essere lenta (il suo limit è 300 s); la risposta dice quanto tempo ci è voluto e se il corridoio è ancora obsoleto. Assegna un nome al corridoio per nome o handle, poiché list_corridors li restituisce.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs

set_cogo_point_udps

Imposta i valori UDP sui punti COGO. 'updates' è un array di oggetti {handle, udp, value, classification?}, da 1 a 500: il valore deve corrispondere al tipo di UDP (una stringa, un numero, un booleano o uno dei valori di un'enumerazione) e 'classification' disambigua un nome UDP condiviso da più di una classificazione. Omettendo dryRun c'è un preview: elenca cosa verrebbe impostato con i valori correnti. Con dryRun false ogni valore viene impostato in una transazione di scrittura.

Ambito: | sources\civil3d\src\Pulse.Civil3D.Server\Tools\WriteTools.cs

Fonte verificata

<https://github.com/LuDattilo/pulse-for-civil3d>

main / 1da6e3cd6a5bd8b2ff4652e0aa7ca88ab0777e8f

<https://pulse-ai.it/guida-civil3d.html>