

URBAN VISUALISATION
& MANAGEMENT GMBH



UVM
SYSTEMS

City**GRID**[®]
2024



Copyright © 2001 - 2024
UVM Systems GmbH

HANDBUCH

Builder

GRID[®]

Inhalt

I.	Grundlagen CityGRID® Builder	5
1	Begriffsbestimmung CityGRID® Builder	5
2	Aufbau eines CityGRID® Builder-Projekts	5
II.	CityGRID® FME Builder	7
1	Installation	7
2	Beschreibung FME Builder	7
2.1	CityGRID® Builder Parameter:	7
2.2	CityGRID® Builder Attribute.....	7
2.3	Tipps für die Verwendung des CityGRID® FME Builders	10
2.3.1	Gibt es bereits bestehende Workspaces zur Datenaufbereitung?	10
2.3.2	Wann muss die Datenklasse gesetzt sein?	11
2.3.3	Welche Geometrietypen werden schnell verarbeitet?	11
2.3.4	Aufbereitung von CityGRID® Gebäude Daten.....	11
2.3.5	CityGML Gebäude Daten.....	12
2.3.6	Beliebige Gebäudedaten mit getrennten Elementen	12
2.3.7	Können beliebig viele Geländekacheln prozessiert werden?	12
2.3.8	Werden die Bildgrenzen einer Geländetextur automatisch erzeugt?	13
2.3.9	Welche Geometriedaten können bei Instanzobjekten verarbeitet werden?	13
2.3.10	Können selbst erstellte Bibliotheksobjekte über FME platziert werden?	13
2.4	Erstellung von Builderprojekten mittels FME Workspaces.....	13
2.4.1	Builder_CGUnits.....	13
2.4.2	Builder_DGM	15
2.4.3	Builder_Punktwolke.....	16
III.	CityGRID® 3D Studio Builder Plug-In	18
1	Installation und Start	18
1.1	Einstellen der Sprache.....	18
1.2	Starten des CityGRID® 3D Studio Builders.....	18
2	Die Benutzeroberfläche	19
2.1	Buttons der Haupt-Toolleiste	20
2.2	Hauptfenster.....	20
2.3	Projekt Explorer	21
2.4	Drag & Drop.....	21
2.5	Festlegen von Tastenschlüsseln (Hotkeys) in 3D Studio MAX	22
2.6	Ändern der Farben von CityGRID® Daten.....	22
3	Arbeiten mit dem CityGRID® 3D Studio Builder	23
3.1	Projekte	23
3.1.1	Neues Projekt erzeugen/ Projekt laden	23
3.1.2	Projekt speichern	25
3.2	Szenen.....	25
3.2.1	Neue Szene anlegen.....	25

3.2.2	Wechseln der aktiven Szene	26
3.3	Anforderungen an 3D Objekte für den CityGRID® Scout.....	26
3.3.1	Geometrie	26
3.3.2	Material.....	26
4	Daten in ein CityGRID® Builder-Projekt einbinden	27
4.1	CityGRID® Daten zu einem Builder-Projekt hinzufügen	27
4.1.1	Auswahlfenster für Datenbankverbindungen öffnen	27
4.1.2	Auswahlfenster für XML Dateien öffnen	27
4.1.3	Arbeitsbereich definieren.....	27
4.1.4	Laden der CityGRID® Daten.....	29
4.2	Fremddaten zu Builder-Projekt hinzufügen.....	30
4.2.1	Anforderungen an Fremddaten	30
4.2.2	Hinzufügen externer Projektdaten	31
4.2.3	Hinzufügen von Projektdaten im 3D Studio	31
4.3	Bibliotheksdaten	32
4.3.1	Aufbau einer Bibliothek.....	32
4.3.2	Neue Bibliothek anlegen	33
4.3.3	Bibliotheksobjekte in Szene einfügen	33
4.3.4	Bibliotheksdaten mittels externer Steuerdatei hinzufügen.....	34
4.3.5	Ändern von Bibliotheksdaten	36
4.4	Geometriedaten aus dem Projekt löschen.....	36
4.4.1	Gelöschte Daten wiederherstellen	36
5	Systemdatenhinzufügen / ändern / entfernen.....	37
5.1	Kamera	37
5.1.1	Neue Kamera erstellen.....	37
5.1.2	Kameraeinstellungen ändern	38
5.2	Sonnenstände	39
5.2.1	Neuen Sonnenstand erstellen	39
5.2.2	Sonnenstand Einstellungen ändern	39
5.3	Systemdaten von Projekt entfernen	40
6	Tools	41
6.1	Objekte auf das Gelände setzen	41
6.2	Objekte vervielfältigen	41
6.2.1	Orthogonalversatz	42
6.2.2	Ringversatz	42
6.2.3	Pfadversatz	42
6.3	Ansicht aktualisieren.....	43
7	Animationen und Bilder rendern.....	44
8	Export zum CityGRID® Scout.....	45
8.1	Steuerung des Exports zum CityGRID® Scout.....	45
8.2	Daten ins Builder-Projekt übernehmen.....	45
IV.	Builder Control Center	47

1	Arten von Builder-Projekten	47
1.1	Einfaches Builder-Projekt	47
1.2	Managed-Builder-Projekt	47
1.3	Vereinigte Builder-Projekte (Gemergte Builder-Projekte)	47
2	Die Builder Control Center Oberfläche	48
3	File	49
4	Manage	49
4.1	Projects	49
4.2	Delete Scenes	50
5	Build	51
5.1	Create	51
5.1.1	Szenen und Varianten	51
5.1.2	Prozessierungsmodi	52
5.1.3	Settings	53
5.1.4	Durchführung der Optimierung	54
5.2	Merge (Vereinen berechneter Scouts)	54
5.2.1	Erstellung eines vereinten Scouts	55
5.3	Config	55
5.3.1	Parameter setzen	56
5.3.2	Verfügbare Parameter	57
5.4	Distribute	60
5.4.1	Export Path	61
5.4.2	Way of Distribution	61
5.4.3	Freischaltung eines Scouts	62
6	Settings	63
6.1	Multi Thread	63
6.2	Manual	63
6.3	Info	63
6.4	Reset	63
V.	Fehlerbehandlung	64
VI.	Kontakt	65

Aufnahme Deckblatt: Prag, Leonhard Niederwimmer, Pixabay

I. Grundlagen CityGRID® Builder

1 Begriffsbestimmung CityGRID® Builder

Der CityGRID® Builder dient zur Zusammenstellung und Optimierung von vorprozessierten 3D Daten für eine performante Echtzeit-Visualisierung mittels CityGRID® Scout. Der Builder besteht aus verschiedenen Komponenten, die entweder eigenständige Applikationen oder Plug-Ins in Fremd-Softwarepakete sind:

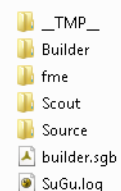
- ein Plug-In in Autodesk 3ds Max, dem „3D Studio Builder Plug-In“
- ein Plug-In in Safe Software FME, dem Writer „CityGRID Builder“
- das Projektverwaltungs-Werkzeug „CityGRID® Builder Control Center“ für die Übersicht und Reorganisation von Daten eines Scout-Projekts, (z.B. Verschieben in Varianten, Löschen von Daten, etc.). Weiters können damit Scouts publiziert (lizenzfrei geschaltet) werden.
- ein Kommandozeilenprogramm, mit dem der Benutzer im Allgemeinen nicht arbeitet, dass jedoch von allen anderen Komponenten verwendet wird, der „builder.exe“

Die Module sind zum Teil aufeinander aufbauend und nur bedingt gegeneinander austauschbar. Zur Generierung eines 3D Projekts werden, je nach Komplexität und Ursprung der Quelldaten unterschiedliche Module des Builder zum Einsatz kommen. Punktwolken, Geländemodelle und CityGRID® Gebäudemodelle etwa können effizient über den FME Builder für den Scout aufbereitet werden, wogegen allgemeine 3D Modelle, Kamerafahrten und Animationen über den 3D Studio Builder in das Scout-Projekt eingespeist werden müssen.

Jedes Datenpaket ist als eine so genannte Szene erstellt. Die Szenen sind innerhalb eines Builder-Projekts eindeutig und sollten stets nur Daten einer Datenklasse (vgl. II 2.2) enthalten. Die Szenen werden dann bei der Optimierung des Scouts (vgl. IV.5.1) aufgelöst und zu komplexen Datenbäumen restrukturiert. Szenen lassen sich sowohl mittels FME Builder als auch 3D Studio Builder generieren und werden bei der Optimierung des Scouts vollkommen gleichrangig behandelt.

2 Aufbau eines CityGRID® Builder-Projekts

Ein CityGRID® Builder Projekt setzt sich aus einer Sammlung von fix vorgegebenen Verzeichnissen zusammen. Für die Integrität eines Builder-Projekts ist der Erhalt der Verzeichnisstruktur essentiell, weshalb manuelle Eingriffe durch den Anwender möglichst unterbleiben sollten. Wenn ein neues Builder-Projekt angelegt wird, so ist ein Zielverzeichnis zu definieren und der Builder übernimmt den Aufbau der Verzeichnisstruktur selbständig.



Jedes Builder-Verzeichnis besitzt außerdem eine Datei builder.sgb. Dies ist die zentrale Verwaltungsdatei des Builder-Projekts, in der globale Einstellungen für das zu erstellende Scout-Projekt gespeichert werden. Mittels dieser Datei lässt sich das Builder-Projekt auch im Builder Control Center öffnen.

In der SuGu.log Datei sind sämtliche Logging – Informationen aus der Erstellung des Builder-Projekts enthalten. Bei Supportanfragen ist diese Datei stets an UVM Systems zu liefern.

Je nach verwendetem Plug-In kann die Ordner Struktur leicht variieren. Stets vorhanden sind die Verzeichnisse „Builder“, „Scout“ und „Source“.

__TMP__: Ist ein Temporärverzeichnis für die Erstellung des Scout-Projekts.

Builder: Dieses Verzeichnis enthält jede Szene, die mittels CityGRID® Builder Plug-In für 3D Studio erstellt wurde. Wenn das Builder-Verzeichnis im 3D Studio geladen wird, wird dieser Ordner ausgewertet und jede darin vorhandene Szene im 3D Studio im Projekt Explorer angezeigt.

ClusterDir: Arbeitsverzeichnis das bei der Kombination mehrerer Builder-Projekte genutzt wird um Datengruppen zu Clustern zusammenfassen zu können.

fme: Hierbei handelt es sich um ein temporäres Arbeitsverzeichnis, das vom FME Builder Plug-In genutzt wird. Im Allgemeinen sollte dieses Verzeichnis nach erfolgreichem Anschluss einer Prozessierung in FME leer sein. Andernfalls wird sich eine Warnung im FME.log finden.

LinkDir: Verwaltungsverzeichnis zur Speicherung der Quellordner unterschiedlicher Builder-Projekte.

ProcessDir: Arbeitsverzeichnis das während der Optimierung eines Scouts mit Daten aus dem Source-Ordner befüllt wird.

Scout: In diesem Verzeichnis befindet sich nach Abschluss aller Aufbereitungs- und Optimierungsarbeiten das fertige Scout-Projekt. Sobald das Verzeichnis mit Daten befüllt wurde, kann dieses Verzeichnis an jeden beliebigen anderen Ort kopiert werden, um die Verteilung des Scouts zu ermöglichen.

Source: Alle prozessierten Daten aus dem FME und 3D Studio Builder Plug-In werden in diesem Verzeichnis vorgehalten. Sobald alle Daten für das Scout-Projekt aufbereitet wurden, kann über FME oder das Builder Control Center die Erstellung des Scouts aus den Source-Daten gestartet werden.

Die Source-Daten können über einen beliebigen Zeitraum sukzessive erstellt werden. Falls gleichnamige Daten im Source-Ordner bzw. den darin enthaltenen Unterordnern vorhanden sind, werden diese bei neuerlicher Ableitung derselben Daten überschrieben.

Alle Daten im Source-Verzeichnis werden stets in nicht reduzierten Koordinaten vorgehalten.

II. CityGRID® FME Builder

1 Installation

Der FME Builder ist technisch gesehen ein Writer in FME, der Daten die an ihn geschickt werden in eine vorgegebene Ordnerstruktur gliedert und für die Optimierung des Scout-Projekts vorbereitet. Um den FME Builder korrekt nutzen zu können ist vor der Installation des CityGRID® Moduls eine unterstützte FME Version am Rechner zu installieren.



Hinweis: Die Liste der unterstützten FME Versionen kann im Handbuch CityGRID® Grundlagen nachgelesen werden. Während des CityGRID® Setups sind weiters alle unterstützten und am Rechner installierten FME Versionen wählbar.



Hinweis: Der CityGRID® FME Builder ist ab FME Base nutzbar, empfohlen ist aber zumindest die Verwendung von FME Professional.

Beim jedem Versionswechsel von FME (etwa beim Umstieg auf eine neue Jahresversion) ist die gesamte FME Komponente von CityGRID® zu deinstallieren und nach der Neuinstallation von FME wieder neu zu installieren.



Hinweis: Kontrollieren Sie in der Änderungsliste des ausgelieferten CityGRID® Setuppaketes ob die zu installierende FME Version auch durch CityGRID® unterstützt wird und kontaktieren Sie im Zweifel den CityGRID® Support.

2 Beschreibung FME Builder

Bei korrekter Installation steht der CityGRID® Builder in der Liste der in FME auswählbaren Writer zur Verfügung. Fügt man ihn einem Workspace hinzu, wird ein Feature Type eingefügt, an den Daten aller Art geschickt werden können. Um die Performance der Datenverarbeitung und Darstellung im CityGRID® Scout zu verbessern sollten die Daten in einem vorgeschalteten Workspace attributiv angereichert werden (vgl. II 2.2)

2.1 CityGRID® Builder Parameter:

1. Der *Destination Folder* des Writers ist ein Verzeichnis, in dem vom CityGRID® Builder eine vordefinierte Unterverzeichnisstruktur angelegt wird. Diese Unterverzeichnisse dürfen nicht manipuliert werden. Man kann auch das Verzeichnis eines bestehenden CityGRID® Builder-Projekts auswählen. Dieses Verzeichnis können sowohl der FME Builder als auch der 3D Studio Builder gemeinsam benutzen um eine komplexe Scout-Projekt mit Rohdaten zu befüllen.



Hinweis: Der Standard-Parameter *Fanout Dataset* wird vom CityGRID® FME Builder nicht ausgewertet.

2. *Generate Scout* weist den FME CityGRID® Builder an, nach Beendigung des FME Workspaces gleich das Builder Control Center zu starten und das soeben aufbereitete Builder-Projekt zu laden, damit direkt die Optimierung des Scouts starten kann.
3. *Instance Library* legt den Pfad zu einer CityGRID® Bibliothek fest, in der vorbereiteten Instanzobjekte zur automatischen Platzierung in einem CityGRID® Scout gefunden werden können. Derartige Bibliotheken können von UVM Systems erworben werden um etwa Baummodelle oder Menschen massenhaft im Scout darstellen zu können.

2.2 CityGRID® Builder Attribute

Neben den Parameter bestimmen nachstehende Standardattribute des CityGRID® FME Builder die Handhabung der geschickten Daten. Attribute die im Laufe eines vorgelagerten Workspaces erzeugt wurden, werden im Allgemeinen nicht ausgewertet und vom FME Builder ignoriert.



Hinweis: Die Attribute des FME Builder sind alle optional und dürfen ungesetzt bleiben. Falls ein oder mehrere Attribute nicht gesetzt sind, greift ein voreingestellter Defaultmechanismus der die geschickten Daten nach gewissen Regeln kategorisiert und in die Daten in die vorgesehenen Unterverzeichnisse einreicht.

1. cgbuilder SceneID:

Jedem Datensatz ist ein eindeutiger Szenen ID zuzuordnen, der der eindeutigen Identifizierung der Daten im Builder-Projekt dient. Dieser darf bis zu 8 Stellen haben und keinen Unterstrich bzw. -Sonderzeichen enthalten. Die Verwendung von Szenen IDs empfiehlt sich bei Visualisierungsprojekten mit vielen Datenquellen um die Übersichtlichkeit der einzelnen Szenebestandteile zu erhöhen und das Projektmanagement zu vereinfachen. Falls kein Szenen ID angegeben wird, wird „00“ verwendet.



Hinweis: Der Projektcode sollte für das gesamte Visualisierungsprojekt eindeutig sein! Falls im Projektcode das Zeichen „_“ enthalten ist, wird es automatisch durch ein beliebiges gültiges Zeichen ersetzt und in FME eine entsprechende Warnung ausgegeben.

2. cgbuilder DatasetName:

Der CityGRID® Builder versucht, den Daten für die Scoutszene „sinnvolle“ Namen zu geben. Falls das Attribut `cgbuilder_DatasetName` gesetzt ist, wird dieses für den Dateinamen verwendet. Insbesondere bei komplexeren Szenen empfiehlt es sich, den einzelnen Daten sinnvolle Namen zu geben, um sie später identifizieren zu können. Falls das Attribut nicht gesetzt ist, werden Namen für die Daten automatisch festgelegt. Wenn im Reader der Workbench das Attribut `fme_dataset` veröffentlicht ist, wird der Dateiname der gelesenen Datei eingebunden.

3. cgbuilder SceneName:

Um Daten zu kapseln, kann das Attribut `cgbuilder_SceneName` gesetzt werden. Wird ein Szenenname vergeben, landen die prozessierten Daten im Source Verzeichnis des Builder-Projekts in einem gleichnamigen Ordner. Jeder angelegte Ordner kann im Builder Control Center zur Definition von Varianten verwendet werden. (vgl. IV 5.1.1) Alle Daten ohne gesetzten Szenennamen werden dem Hauptdatensatz (z.B. Umgebungsmodell) zugeordnet und sind im Scout stets sichtbar und daher auch keiner Variante zuordenbar.



Tipp: Planungsvarianten werden eingesetzt, um verschiedene Planungszustände in Echtzeit ohne Performanceverlust wechseln zu können. Zu diesem Zweck müssen die Daten aller Varianten vorprozessiert werden und jeweils in eigenständige Planungsvarianten abgelegt werden. Zur Laufzeit eines Scout-Projektes werden beim Wechsel einer Planungsvariante alle Daten der geladenen Variante entladen und durch die Daten der zu ladenden Variante ersetzt. Daten, die sich zur Laufzeit nicht verändern sollen, werden als Bestandsdaten bezeichnet und keiner Planungsvariante zugeordnet.

4. cgbuilder DataClass:

Je nach Datenklasse wendet der Builder unterschiedliche Strategien an, um verschiedene „Level of Detail“ für das Scout-Projekt zu generieren und um die Daten in einem Baum einzuordnen. Diese LoD Stufen sind von den LoD Stufen des Gebäudes unabhängig und weisen wesentlich mehr Stufen auf. Je nach Entfernung eines Objekts in der Szene lädt der Scout unterschiedliche LoDs um performanceoptimiert zu arbeiten. Dabei werden je nach Datenklasse Vereinfachungsoperationen an der Geometrie bzw. dem Bildmaterial vorgenommen oder Objekte gänzlich ausgeblendet. Für eine gute Performance des Scouts bei großen Datenmengen ist daher die korrekte Erkennung bzw. Angabe der Datenklasse essenziell. Das Attribut kann folgende Werte annehmen:

- IMAGE

Wird für alle Bilddaten, die über FME an den CityGRID® Builder geschickt werden, gesetzt. Insbesondere georeferenzierte Bilddaten zur Geländetexturierung (z. B. Orthofotos) sind Kandidaten für die Zuweisung dieses Attributes.

- STATICBUILDING

Alle semantisch korrekten CityGRID® und CityGML Gebäude fallen in diese Klasse. Gegenüber anderen 3D Daten zeichnen sich Objekte dieser Datenklasse dadurch aus, dass sie getrennt in

Dach und Fassade (bzw. allfällige weitere Elemente) vorkommen, jedoch über eine attributive Beschreibung zu Gebäuden zusammengefasst werden können. Bei der LoD Ableitung wird das Gebäude geometrisch immer mehr vereinfacht und kann im Extremfall auch komplett entfernt werden (z. B. kleine Objekte bei sehr großer Entfernung).



Hinweis: Die Charakteristik der Daten in der Klasse Static Building muss stets einem Hauscharakter entsprechen. Der angewandte Optimierungsalgorithmus ist auf diese Form hin optimiert.

- **STATICTERRAIN**

Umfasst alle 2,5D Daten die Gelände oder Oberflächenmodelle darstellen. Falls Texturen verwendet werden, sollte ein regelmäßiges, weitestgehend gleichmaschiges Mosaik von Bildern (z. B. Orthofotos) zum Einsatz kommen.

- **STATICDESIGN**

Alle 3D Daten, die weder der Static Building noch der Static Terrain Datenklasse zuzuweisen sind, finden sich hier. Im Gegensatz zu den beiden anderen Datenklassen können Daten vom Typ Static Design den speziellen Optimierungsalgorithmus zur Aufbereitung der LoD Stufen nicht nutzen, wodurch bei sehr großen Szenen die Performance der Szene beeinträchtigt sein kann. Dafür ist wiederum sichergestellt, dass alle Daten, auch in niedrigen LoD Stufen, stets vorhanden sind und nie entfernt werden.



Hinweis: Alle 3D Daten, die weder Geländemodelle noch Gebäude darstellen, sind dieser Klasse zuzuschlagen (Straßen, Eisenbahn, Denkmäler etc.).

- **STATICPOINT**

Alle Punktwolken sind durch Setzen des Attributs zu dieser Datenklasse zuzuweisen.

- **STATICINSTANCE**

Wiederkehrende Daten einer 3D Szene können speicheroptimiert mittels Instanzobjekten (Bibliotheksobjekten) dargestellt werden. Durch Kombination einer attributierten Punktliste, aus der die Einsetzpunkte gewonnen werden und dem Link zu einer CityGRID® 3D Bibliothek, baut der CityGRID® FME Writer die 3D Objekte im Scout auf.

Falls der Datentyp nicht gesetzt ist, werden folgende Regeln angewandt:

- Falls die Daten von einem CityGRID® Reader stammen, kann der Builder die Datenklasse automatisch erkennen. Es werden insbesondere die Reader-Attribute *citygrid_ElementClass* und *citygrid_ObjectClass* sowie *citygrid_BuildingElement_id* etc. verwendet.
- Auch für semantisch korrekte CityGML Daten kann der Builder die Datenklasse automatisch erkennen. Daher können auch CityGML Daten direkt an den Builder geschickt werden, wobei insbesondere der Level of Detail des Gebäudemodells für die Scout-Optimierung verwendet wird.



Tipp: Bei sehr großen Datenmengen empfiehlt es jedoch, zuerst mittels des CityGRID® FME Writers eine CityGRID® XML Datei zu erzeugen, da CityGRID® Daten noch besser optimiert werden können.

- Für Punktdaten und Bilddaten wird die Datenklasse automatisch aus dem Geometrietyp erkannt.
- Instanzdaten können nicht automatisch erkannt werden.
- Andere Daten werden als „static design data“ interpretiert.

5. cgbuilder_OmitUntextured:

Regelt das Verhalten von Geländemodellflächen ohne Bildüberdeckung. Durch die fehlende Überdeckung kann keine Textur aufgebracht werden, was zu unschönen Effekten in der Visualisierung führen kann. Wenn Features von Geländemodellen das Attribut *cgbuilder_OmitUntextured* gesetzt haben, tritt folgender Mechanismus in Kraft

- Wert = 1:
Alle untexturierten Dreiecke werden ausgesondert und in eine eigene Datei im „fme“ Verzeichnis des Builder-Projekts gespeichert. Für die Scout-Erstellung werden nur die texturierten Teile des Geländemodells herangezogen, die untexturierten Flächen verbleiben für eine etwaige genauere Untersuchung im fme Ordner. Im Log-File von FME findet sich eine entsprechende Warnung, dass eine Datei mit allen untexturierten Flächen des Geländemodells gebildet wurde.
 - Wert = 0 oder Attribut ist nicht vorhanden:
Alle Dreiecke des Geländemodells werden an den Scout geschickt, egal ob Texturen zugewiesen wurden oder nicht.
6. Cginstance Length:
Definiert die Länge des zu platzierenden Instanzobjekts
 7. cginstance Width:
Legt die Breite des Instanzobjekts fest
 8. cginstance Height:
Bestimmt die Höhe des Instanzobjekts
 9. cginstance Diameter:
regelt bei achssymmetrischen Objekten (z.B.: Bäumen) den Durchmesser des Instanzobjekts
 10. cginstance Library:
Definiert den Bibliotheksnamen für das Instanzobjekt
 11. cginstance Category:
Legt den Kategorienamen eines Instanzobjekts fest
 12. cginstance Name:
bestimmt den Namen des zu platzierenden Instanzobjekts.



Beispiel: In der CityGRID® Baumbibliothek mit Namen „Vegetation“ liegt in der Kategorie „Bionatics“ das Baummodell „Ahorn“. Wenn der Baum 20m hoch und einen Krondendurchmesser von 8m haben soll, sind die `cginstance_Attribute` folgendermaßen zu setzen:

`cginstance_Height: 20`

`cginstance_Diameter: 8`

`cginstance_Library: Vegetation`

`cginstance_Category: Bionatics`

`cginstance_Name: Ahorn`



Hinweis: Wenn keine Attribute für Länge, Breite, Höhe oder Durchmesser gesetzt sind, platziert der Builder die Bibliotheksobjekte mit ihrer Originalgröße des Bibliotheksobjekts im Scout. Andernfalls wird das Objekt entsprechend skaliert. Ist lediglich ein Wert definiert erfolgt die Skalierung proportional, andernfalls in jeder angegebenen Dimension.

Bei Nutzung des Diameter Parameters darf nur noch der Height Wert gesetzt sein, da der Diameter Länge und Breite gleichermaßen skaliert.

2.3 Tipps für die Verwendung des CityGRID® FME Builders

Im folgenden Abschnitt finden sich einige typische Fragestellungen sowie Lösungsvorschläge zum optimalen Einsatz des CityGRID® FME Builders.

2.3.1 Gibt es bereits bestehende Workspaces zur Datenaufbereitung?

Für die gängigsten Datensituationen sind im Installationsverzeichnis von CityGRID® in den Unterverzeichnissen „FME Plugins\Builder“ (z.B. C:\Program Files\CityGRID\FME Plugins\Builder) Workspaces bereitgestellt. Mit

diesen Workspaces lassen sich CityGRID® Units und Geländemodelle, CityGML Gebäude sowie Punktwolkendaten für die Nutzung im CityGRID® Builder aufbereiten.



Hinweis: Die Workspaces werden bei jedem Setup überschrieben. Es ist daher ratsam Kopien der Datei an einem anderen Ort anzulegen und mit den Kopien zu arbeiten. Die Workspaces im Installationsverzeichnis dienen als Kopiervorlage des jeweils aktuellen Zustandes der Workspaces.

2.3.2 Wann muss die Datenklasse gesetzt sein?

Während für Punkt- und Bilddaten die Datenklasse automatisch aus dem Geometrietyp erkannt werden kann, ist für Flächendaten eine Unterteilung nur anhand von Attributen möglich. Daher sollte die *cgbuilder_DataClass* unbedingt gesetzt werden für:

- Geländemodelle, die nicht von einem CityGRID® oder CityGML Reader kommen (z.B. durch einen SurfaceModeller generiert). Derart generierten Modellen ist die Datenklasse *STATICTERRAIN* zuzuweisen.
- Gebäudedaten, die nicht von einem CityGRID® oder CityGML Reader kommen, müssen der Datenklasse *STATICDESIGN* zugewiesen werden. Zusätzlich sollte das Attribut *citygrid_ElementClass* auf „Roof“, „Facade“ oder „Floor“ gesetzt werden, wenn die Geometrie nach diesen Elementen getrennt vorhanden ist. Dadurch wird bei untexturierten Gebäuden die Standardfarbe im Scout für Dach, Fassade und Boden unterschiedlich gesetzt (siehe später).

2.3.3 Welche Geometrietypen werden schnell verarbeitet?

Punktwolken sollten als IFMEPointCloud geschickt werden.

Geländemodelle als IFMEMesh mit Dreiecken als grundlegenden Geometrieobjekten. Der CityGRID® FME Builder kann auch mit beliebigen Polygonen in einem Mesh umgehen, allerdings muss jedes Polygon durch den CityGRID® Builder erst wieder in eine Dreiecksvermaschung zerlegt werden, was je nach Flächenanzahl sehr zeitintensiv werden kann.

Ebenso können auch Aggregate aller Punkte bzw. aller Geländedreiecke (z.B. IFMECompositeSurface) verwendet werden, wobei jedoch die Verarbeitungszeit höher ist.

2.3.4 Aufbereitung von CityGRID® Gebäude Daten

Während der Writer für CityGRID® ein eigenes Feature für die Unit (mit den Unit-Attributen) und für das Element (mit den Element-, Elementkomplex-, und Objektattributen) erwartet, hat der Builder nur einen einzigen Writer-Port. An diesen werden in erster Linie die Surface Daten eines CityGRID® Readers geschickt.



Hinweis: Der Builder hat nicht die Funktionalität die Flächen zu erzeugen. Um die Triangulation eines Liniengerüsts im CityGRID® Schema zu bewirken ist der CityGRID® Administrator, Modeler oder FME Writer nötig.

Um Element-Attribute dem Builder zu übergeben (z.B. die Elementklasse für eine automatisierte Einfärbung der Elemente je nach Klasse), können die Attribute mit dem Transformer FeatureMerger auf die jeweiligen Surface-Features übertragen werden. Ebenso können auch die Element-Features zusammen mit den Surface Features direkt an den Builder geschickt werden, wobei in diesem Fall neben den Attributen *citygrid_Building*_id* auch *cgbuilder_DataClass*, *cgbuilder_SceneName* und *cgbuilder_SceneID* bei allen Features gleich sein müssen (falls sie gesetzt sind).



Tipp: Das Schicken des Element Features an den FME Builder empfiehlt sich nur dann, wenn außer CityGRID® Daten keine weiteren Daten im selben FME Prozess an den Builder geschickt werden. Falls unterschiedliche Datensätze gemeinsam mit nur einem Builder-Aufruf prozessiert werden, ist die Verwendung eines Feature Mergers angeraten, um alle Attribute an die Surface-Features zu hängen.



Hinweis: Liniendaten brauchen an den Builder nicht geschickt werden, da sie im Scout auch nicht dargestellt werden.

2.3.5 CityGML Gebäude Daten

Semantisch korrekte CityGML Daten werden vom CityGRID® Builder bestmöglich automatisch interpretiert. Zu beachten ist:

- Falls in den CityGML Daten eine Terrain Intersection Curve (TIC) vorhanden ist, kann es durch die Behandlung in FME zur (unerwünschten) Bildung einer Fläche kommen, was visuelle Beeinträchtigungen im Scout zur Folge hat. Um dies zu verhindern, empfiehlt es sich die TIC nicht an den FME Builder zu senden.
- Der FME CityGML Reader liest stets nur das erste Texturthema ein und ignoriert allfällige weitere. Falls Textur nicht auf dem ersten Texturthema zu liegen kommt, wird diese derzeit auch nicht in FME eingelesen werden, und auch nicht in die Scout-Projekte gelangen.

2.3.6 Beliebige Gebäudedaten mit getrennten Elementen

Falls Gebäudedaten nicht von CityGRID® oder CityGML stammen, gibt es keine allgemeinen Regeln, welche Flächen zu einem Gebäude oder zu einem Element (Dach, Fassade) zusammengefasst werden können. Dies muss der Anwender in der Workbench durch Gruppieren und Setzen von Attributen machen. Dazu sollte er folgende Attribute generieren:

- *citygrid_DataClass* auf STATICBUILDING
- *citygrid_FeatureType*: mit dem Wert „Surface“, um dem Builder mitzuteilen, dass es sich um Flächen eines Gebäudeteils handelt
- *citygrid_BuildingUnit_id*, *citygrid_BuildingObject_id*, *citygrid_BuildingElementComplex_id* und *citygrid_Element_id* auf eindeutige Werte je nach Komplex, um die CityGRID®-Gebäudehierarchie zu simulieren.



Beispiel: Einem Gebäude mit einem Hauptobjekt und einer Garage sollen die vier *citygrid_BuildingIds* zugeordnet werden. Dazu ist folgende Attributierung vorzunehmen:

Allen Features des Gebäudes wird die *citygrid_BuildingUnit_id* 1 zugewiesen, da sie ja alle Teil des Gebäudes sind.

Alle Features des Hauptgebäudes bilden im Sinne der CityGRID® Logik ein Objekt und erhalten daher den *citygrid_BuildingObject_id* 1 bzw. jene der Garage den Wert 2

Da keine weitere Unterteilung der Gebäude (Liftanbau, Rampe etc.) gegeben ist, kann der *citygrid_BuildingObject_id* gleich dem *citygridBuildingElementComplex_id* sein

Die Einzelflächen des Hauptgebäudes sollten sich nun noch in Dach, Fassade und Boden (falls vorhanden) gliedern und den Features der jeweiligen Elementklasse das *citygrid_BuildingElement_id* Attribut mit den Werten 1,2 oder 3 zugewiesen werden.

Dadurch ergibt sich für alle Features der Fassadenflächen des Hauptgebäudes eine Attributabfolge von 1/1/1/2 bzw. für den Boden des Garagenobjekts 1/2/2/3



Hinweis: Je nach Datenlage ist das Generieren der genannten *citygridBuilding..ids* nicht immer möglich bzw. nur bis zu einem gewissen Hierarchiegrad. Es empfiehlt sich die Strukturierung stets so weit wie möglich zu betreiben, um die Aufbereitung der Visualisierungs-LoDs bestmöglich zu unterstützen. Das Scout-Projekt wird auch bei gänzlich fehlender Hierarchisierung der externen Daten entstehen, allerdings sind Performanceprobleme umso wahrscheinlicher, je weniger strukturiert die Daten an den Scout geschickt werden.

- *citygrid_ElementClass* auf „Roof“, „Facade“ oder „Floor“, falls Standardfarben je nach Elementklasse zugewiesen werden sollen.

2.3.7 Können beliebig viele Geländekacheln prozessiert werden?

Wie jeder Computerprozess ist auch die Datenaufbereitung mittels CityGRID® Builder an Speichergrenzen gebunden. Je nach Rechnerkonfiguration können dadurch unterschiedlich große Datenmengen verarbeitet werden. Sobald die Speichergrenzen aber überschritten sind, kommt es zum Abbruch des Vorganges, der sich im Allgemeinen durch einen so genannten „Bad allocation“ Fehler bemerkbar macht. Um diesen Speicherüberlauf zu vermeiden, steht nur die Aufteilung der prozessierten Daten in mehrere Prozesse zur Verfügung. Wenn der

Scout aus den Builder Rohdaten zusammengesetzt wird, ist die Menge der Daten aber keine limitierende Größe, da durch die Vorprozessierung im Builder bereits eine optimierte Datenstruktur aufgebaut wurde.



***Tipp:** Bei großen Datenmengen empfiehlt es sich, iterativ vorzugehen und zunächst mehrere Erstellungsprozesse mit jeweils überschaubarer Datenmenge zu bearbeiten. Durch sukzessives Steigern der verarbeiteten Daten kann so die individuelle Höchstmenge an zu verarbeitenden Daten ermittelt werden.*

2.3.8 Werden die Bildgrenzen einer Geländetextur automatisch erzeugt?

Geländetextur kann von sowohl von einem CityGRID® Reader, als auch von Tiff oder Jpg oder ähnlichen Readern (wenn eine World-Datei vorhanden ist) verwendet werden. Der Builder rechnet automatisch Bildgrenzen in die Triangulierung ein. Um die volle Schärfe als Geländetextur zu erhalten, dürfen Bilder maximal 4096x4096 Pixel haben. Dies lässt sich mit einem vorgeschalteten RasterTiler in FME erzielen.

2.3.9 Welche Geometriedaten können bei Instanzobjekten verarbeitet werden?

Instanzobjekte benötigen einen Einsetzpunkt, um im Scout platziert zu werden. Dieser Einsetzpunkt wird aus Punktfeatures extrahiert, die an den CityGRID® FME Writer geschickt werden. Andere Datenklassen als Punkt lassen sich nicht für Instanzobjekte nutzen. Wenn die Punkte 3D Koordinaten aufweisen, übernimmt der Writer diese Werte und platziert die Objekte entsprechend. Liegen keine 3D Werte vor, so sind an den Writer 2D Punktfeatures zu senden, die Instanzobjekte werden dann auf Höhe 0 platziert. Mittels Builder Control Center steht eine Möglichkeit zur Verfügung, Instanzobjekte in der Höhe zu verändern, in dem die Punkte auf Flächen von anderen Scout-Projekten gehoben werden (vgl. IV 5.3)

2.3.10 Können selbst erstellte Bibliotheksobjekte über FME platziert werden?

Dies ist nicht möglich, da individuell erstellte Objekte als .max Dateien in der Bibliothek gespeichert sind (vgl. III 4.3). Das proprietäre .max Format lässt sich mittels FME nicht verarbeiten.

2.4 Erstellung von Builderprojekten mittels FME Workspaces

Zum Aufbau von Builder-Projekten stellt UVM Systems ein Set an vordefinierten Workspaces bereit, mittels der sich die gebräuchlichsten Datentypen verarbeiten lassen. Zum Einsatz der Workspaces sind die entsprechenden publizieren Parameter mit Werten zu befüllen, eine Adaptierung der Programmlogik ist im Allgemeinen nicht nötig. Allfällige Anpassungsarbeiten werden auf Kundenwunsch durch UVM Systems, im Rahmen des CityGRID® Supports angeboten. Bitte wenden Sie sich im Bedarfsfall an den CityGRID® Support.



***Hinweis:** Die nachstehenden Workspaces sind Teil des CityGRID® Setups und werden im Installationsverzeichnis unter FME Plugins\Builder kopiert. Diese Workspaces sollten als Kopiervorlage genutzt werden. Eine Bearbeitung im Installationsverzeichnis ist nicht zielführend und sollte unterlassen werden.*

2.4.1 Builder_CGUnits

Aufgabe dieses Workspaces ist es CityGRID® Units in ein Builder-Projekt zu überführen. Von den Units werden nur die Flächen ausgewertet, der Liniengerüst kann im Scout nicht verarbeitet werden und wird daher vom Workspace nicht ausgewertet. Der Workspace nimmt keine Veränderungen an den Flächen vor, somit sind die Units durch entsprechende Vorverarbeitungsschritte in die gewünschte Form zu bringen.



***Hinweis:** Die Units müssen über triangulierte Flächen verfügen, um verarbeitet werden zu können. Folglich ist es unerheblich, ob es sich um Units mit Liniengerüst handelt, oder Flächendaten mit Flächenbildungstyp „eingefrorenes Flächennetz“ handelt.*

Folgende Parameter sind für den ordnungsgemäßen Betrieb zu setzen. Die Werte in [] stellen die Parameter für den Batch-Betrieb des Workspaces dar:

- Kachel: [p_TILE]

Optionaler Parameter um wiederkehrende Bezeichnungen, wie den Kachel-ID einer Blattschnittzelle, als Variable setzen zu können. Dieser Wert ist als Bestandteil weiterer Parameter des Workspaces anwendbar. Um auf den Wert zugreifen zu können ist der Verweis „\$(p_Tile)“ zu nutzen.

- **Modellname (getrennt durch ';'):** [MODEL_NAMES]

Optionaler Parameter um Modelle aus einer CityGRID® Datenbank laden zu können. Im Gegensatz zu XML Dateien, die stets nur ein einziges Modell enthalten dürfen, können Datenbanken eine beliebige Anzahl an Modellen aufweisen. Durch Angabe des Modellnamens wird der CityGRID® FME Reader angewiesen exakt jenes Modell aus der Datenquelle zu laden. Bei der Nutzung von XML Dateien als Datenquelle sollte dieser Parameter leer bleiben.

- **Input CityGRID Gebäude:** [S_CITYGRID]

Pflichtparameter mit dem Pfad zu CityGRID® XML Datei(en) mit Units- Alternativ kann auch der Pfad zu einer CityGRID®-ini Datei mit den Anmeldeparametern zu einer CityGRID® Datenbank verwendet werden. Falls mehrere XML Dateien gelesen werden sollen, kann die Wildcard „*“ anstatt des Dateinamens genutzt werden.

- **Texturen verwenden:** [READ_IMAGES]

Pflichtparameter zur Auswahl der Oberflächenbeschaffenheit. Es lassen sich folgende Werte setzen:

- **Ja:**

Die Texturen von Units werden an das Builderprojekt weitergeleitet und bei der Scouterstellung genutzt. Allfällig untexturierte Flächen erhalten die CityGRID® Standardfarben

- **Nein:**

Allfällig vorhandene Texturen werden nicht ausgelesen und die Flächen der Units werden nach den Vorgaben des Parameters „Flächenfarben“ eingefärbt.

- **Verzeichnis Builderprojekt:** [D_CITYGRIDBUILDER]

Pflichtparameter um den Speicherort des Builder-Projekts zu definieren. Unter diesem Pfad legt der CityGRID® FME Builder die Ordnerstruktur des Builder-Projekts an (vgl. I 2), und speichert die durch den Workspace aufbereiteten Daten ab. Dieses Verzeichnis enthält auch die builder.sgp Datei, mittels der das Projekt im Builder Control Center geladen werden kann, um den Scout zu erstellen. (vgl. VI 3)

- **Flächenfarben:** [p_COLOR]

Pflichtparameter zur Auswahl der Flächenfärbung, falls keine Texturen von Units ausgelesen werden sollen. Es lassen sich folgende Werte setzen:

- **CG Standard:**

Allfällig vorhandene Texturen werden nicht ausgelesen und die Flächen der Units mit den CityGRID® Standardfarben eingefärbt.

- **Individuell:**

Allfällig vorhandene Texturen werden nicht ausgelesen und die Flächen der Units werden mit individuell gesetzten Farbwerten eingefärbt. Dabei wird zwischen Dachflächen und allen übrigen Flächen unterschieden. Der Farbton für Dach und die übrigen Flächen, zusammengefasst unter „Fassaden“, lässt sich in den privaten Parametern des Workspaces setzen.

- **Einheitsfarbe:**

Allfällig vorhandene Texturen werden nicht ausgelesen und die Flächen der Units erhalten einen einheitlichen Farbton. Der Einheitsfarbton lässt sich in den privaten Parametern des Workspaces setzen.

- **Szene:** [p_SCENE_NAME]

Pflichtparameter um das aufbereitete Datenpaket innerhalb eines Builder-Projekts ansprechen zu können. Daten einer Szene stellen zusammengehörende Daten einer Datenklasse dar (z.B. alle Units einer Blattschnittzelle). Dieses Datenpaket lässt sich im Builder Control Center, während des Create Prozesses auswerten und in die Scout-Erstellung ein-/ausschließen.



***Hinweis:** Szenen dienen der Organisation der Builder-Rohdaten. Während der Scout-Erstellung geht die Information über die Szene verloren*

- **Dateiname:** [p_DATASET_NAME]

Pflichtparameter zur Bezeichnung der aufbereiteten Dateien im Builder-Projekt. Der Dateiname muss je Szene individuell festgelegt werden, um das Überschreiben von bereits aufbereiteten Daten zu verhindern.



***Tipp:** Für die Parameter Szene und Dateiname lässt sich der optionale Parameter Kachel heranziehen. Durch Setzen von \$(p_Tile) bei Szene und Dateiname wird der Wert von Kachel übergeben. Derart bedarf es nur einer Änderung des Werts von Kachel um sicher zu stellen, dass Szene und Dateiname mit gleichen, eindeutigen Werten gesetzt sind.*

- **Log File:** [LOG_FILE]

Pflichtparameter um den Speicherort der FME-Protokolldatei festzulegen. Zu beachten ist, dass der Pfad zur Log-Datei bereits existieren muss, FME erzeugt keine fehlenden Unterverzeichnisse eines manuell gesetzten Log-Pfades.

2.4.2 Builder_DGM

Aufgabe dieses Workspaces ist es Geländemodelle in ein Builder-Projekt zu überführen. Als Input akzeptiert der Workspace sowohl Geländemodelle im CityGRID® Datenformat, als auch Geländedaten in den Formaten DWG, DGN, Shapefile, ASCIIGrid oder FFS. Zusätzlich bietet der Workspace die Möglichkeit die Geländedaten mittels georeferenzierter Rasterdaten (Worldfile) zu texturieren.



***Hinweis:** CityGRID® Geländemodelle, sowie alle anderen Daten, die bereits als TIN Oberfläche in den Workspace eingebracht werden, lassen sich direkt verarbeiten. Alle anderen Daten unterzieht der Workspace einer Geländetriangulation, die dann an den Builder geleitet wird.*

Folgende Parameter sind für den ordnungsgemäßen Betrieb zu setzen. Die Werte in [] stellen die Parameter für den Batch-Betrieb des Workspaces dar:

- **Kachel:** [p_TILE]

Optionaler Parameter um wiederkehrende Bezeichnungen, wie den Kachel-ID einer Blattschnittzelle, als Variable setzen zu können. Dieser Wert ist als Bestandteil weiterer Parameter des Workspaces anwendbar. Um auf den Wert zugreifen zu können ist der Verweis „\$(p_Tile)“ zu nutzen.

- **Input DTM:** [S_TIN]

Pflichtparameter der den Pfad zu den Geländemodell-Ausgangsdaten enthält.

- **Format DTM:** [S_FORMAT_TIN]

Pflichtparameter der das Dateiformat der zuvor eingelesenen Geländemodell-Ausgangsdaten bestimmt. Nur wenn die Kombination beider Parameter korrekt eingestellt ist, lassen sich die Daten einlesen.

- **Input Geländetextur:** [S_IMAGE]

Pflichtparameter um den Pfad zu Texturbildern für die Geländetextur festzulegen. Falls mehrere Bilder in einem Verzeichnis verwendet werden sollen, kann die Wildcard „*“ als Joker eingesetzt werden.



***Hinweis:** Wenn keine Geländetextur gewünscht ist, so ist ein ungültiger Pfad, z.B. „nicht verwendet“ zu setzen. Der Workspace lässt sich nur starten, wenn dieser Parameter mit einem String beschickt wurde.*

- **Format Geländetextur:** [S_FORMAT_IMAGE]

Pflichtparameter der das Dateiformat der zuvor eingelesenen Geländetextur bestimmt. Nur wenn die Kombination beider Parameter korrekt eingestellt ist, lassen sich die Daten einlesen.

- **Einsetzpunkt Worldfile:** [p_INSERT_POINT]

Pflichtparameter der Lagerung des Einsetzpunktes in der Georeferenzierungsdatei des Bildes (Worldfile) definiert. Standardgemäß ist die Lagerung stets der Mittelpunkt der linken oberen Zelle. Im Falle eines

nicht standardkonformen Worldfiles, lässt sich die Lagerung auch auf den Eckpunkt festlegen. Der Workspace korrigiert die Lage dann selbständig um ein halbes Bodenpixel.



Hinweis: Bei den Koordinaten des Worldfiles ist darauf zu achten, dass sowohl die Auflösung der Pixelgröße am Boden, als auch die Koordinaten des Einsetzpunktes auf exakt drei Nachkommastellen abgegeben sind. Gegebenenfalls ist das Bild vor der Verarbeitung im Workspace zu resampeln bzw. die Lagerung anzupassen.

Falls die Geländetextur aus einem blattschnittfreien Datenbestand abgeleitet wird, sollte der Ausschnitt stets auf Meter genau geschnitten werden. Das freie Aufziehen eines Interessensbereiches führt im Allgemeinen nicht zu der geforderten Koordinatenschärfe beim Einsetzpunkt. Bei der Verarbeitung solcher Daten treten an den Rändern eine Vielzahl an untexturierten Dreiecksflächen auf, da sich die Texturkoordinaten nicht genau bestimmen lassen.

- **Verzeichnis Builderprojekt:** [D_CITYGRIDBUILDER]

Pflichtparameter um den Speicherort des Builder-Projekts zu definieren. Unter diesem Pfad legt der CityGRID® FME Builder die Ordnerstruktur des Builder-Projekts an (vgl. I 2), und speichert die durch den Workspace aufbereiteten Daten ab. Dieses Verzeichnis enthält auch die builder.sgp Datei, mittels der das Projekt im Builder Control Center geladen werden kann, um den Scout zu erstellen. (vgl. VI 3)

- **Szene:** [p_SCENE_NAME]

Pflichtparameter um das aufbereitete Datenpaket innerhalb eines Builder-Projekts ansprechen zu können. Daten einer Szene stellen zusammengehörende Daten einer Datenklasse dar (z.B. alle Geländemodelle einer Blattschnittzelle). Dieses Datenpaket lässt sich während des Create Prozesses im Builder Control Center, auswerten und in die Scout-Erstellung ein-/ausschließen.



Hinweis: Szenen dienen der Organisation der Builder-Rohdaten. Während der Scout-Erstellung geht die Information über die Szene verloren

- **Dateiname:** [p_DATASET_NAME]

Pflichtparameter zur Bezeichnung der aufbereiteten Dateien im Builder-Projekt. Der Dateiname muss je Szene individuell festgelegt werden, um das Überschreiben von bereits aufbereiteten Daten zu verhindern.



Tipp: Für die Parameter Szene und Dateiname lässt sich der optionale Parameter Kachel heranziehen. Durch Setzen von \$(p_Tile) bei Szene und Dateiname wird der Wert von Kachel übergeben. Derart bedarf es nur einer Änderung des Werts von Kachel um sicher zu stellen, dass Szene und Dateiname mit gleichen, eindeutigen Werten gesetzt sind.

- **Flächennormalen wenden:** [p_Switch_Normals]

Pflichtparameter zur Festlegung, ob der Umlaufsinn aller Dreiecke des Geländemodells geändert werden soll. Im Scout sind nur jene Flächen bei einseitiger Flächendarstellung sichtbar, die bei Betrachtung von oben gegen den Uhrzeigersinn gezeichnet wurden.

- **Untexturierte Flächen entfernen:** [p_Omit_Untextured]

Pflichtparameter der das Verhalten für Geländedreiecke ohne Textur steuert.

- **Log File:** [LOG_FILE]

Pflichtparameter um den Speicherort der FME-Protokolldatei fest-zulegen. Zu beachten ist, dass der Pfad zur Log-Datei bereits existieren muss, FME erzeugt keine fehlenden Unterverzeichnisse eines manuell gesetzten Log-Pfades.

2.4.3 Builder_Punktwolke

Aufgabe dieses Workspaces ist es Punktwolkendaten in ein Builder-Projekt zu überführen. Als Input akzeptiert der Workspace ausschließlich LAS oder LAZ Daten. Zusätzlich bietet der Workspace die Möglichkeit die Punktwolken mittels georeferenzierter Rasterdaten (Worldfile) zu färben.



Hinweis: Die Darstellung von Punktwolken ist nur dann möglich, wenn die Punktverteilung einem aktuell gebräuchlichen ALS-Flug entspricht. Punktdichten bis zu 60 Punkte/m² sind problemlos darstellbar und

hinsichtlich der Datenmenge einzig durch die Festplattenkapazität beschränkt. Punktwolken aus Nahbereichsscans hingegen eignen sich nicht für die Verarbeitung, da die Punktdichte zu hoch ist.

Folgende Parameter sind für den ordnungsgemäßen Betrieb zu setzen. Die Werte in [] stellen die Parameter für den Batch-Betrieb des Workspaces dar:

- **Kachel:** [p_TILE]
Optionaler Parameter um wiederkehrende Bezeichnungen, wie den Kachel-ID einer Blattschnittzelle, als Variable setzen zu können. Dieser Wert ist als Bestandteil weiterer Parameter des Workspaces anwendbar. Um auf den Wert zugreifen zu können ist der Verweis „\$(p_Tile)“ zu nutzen.
- **Punktwolken LAS File(s):** [S_LAS]
Pflichtparameter zum Speicherort der Punktwolkendaten. Es lassen sich Daten im Format LAS und LAZ (gepacktes LAS) verarbeiten. Falls mehrere Punktwolkendateien in einem Verzeichnis verwendet werden sollen, kann die Wildcard „*“ als Joker eingesetzt werden.
- **Input Orthofoto:** [S_ORTHO]
Pflichtparameter um den Pfad zu Texturbildern für die Einfärbung der Punktwolkendaten festzulegen. Falls mehrere Bilder in einem Verzeichnis verwendet werden sollen, kann die Wildcard „*“ als Joker eingesetzt werden.



***Hinweis:** Wenn keine Einfärbung gewünscht ist, so ist ein ungültiger Pfad, z.B. „nicht verwendet“ zu setzen. Der Workspace lässt sich nur starten, wenn dieser Parameter mit einem String beschickt wurde.*

- **Format Geländetextur:** [S_FORMAT_ORTHO]
Pflichtparameter der das Dateiformat der zuvor eingelesenen Orthofotos bestimmt. Nur wenn die Kombination beider Parameter korrekt eingestellt ist, lassen sich die Daten einlesen.
- **Verzeichnis Builderprojekt:** [D_CITYGRIDBUILDER]
Pflichtparameter um den Speicherort des Builder-Projekts zu definieren. Unter diesem Pfad legt der CityGRID® FME Builder die Ordnerstruktur des Builder-Projekts an (vgl. I 2), und speichert die durch den Workspace aufbereiteten Daten ab. Dieses Verzeichnis enthält auch die builder.sgp Datei, mittels der das Projekt im Builder Control Center geladen werden kann, um den Scout zu erstellen. (vgl. VI 3)
- **Szene:** [p_SCENE_NAME]
Pflichtparameter um das aufbereitete Datenpaket innerhalb eines Builder-Projekts ansprechen zu können. Daten einer Szene stellen zusammengehörende Daten einer Datenklasse dar (z.B. alle Punktwolken einer Blattschnittzelle). Dieses Datenpaket lässt sich während des Create Prozesses im Builder Control Center, auswerten und in die Scout-Erstellung ein-/ausschließen.



***Hinweis:** Szenen dienen der Organisation der Builder-Rohdaten. Während der Scout-Erstellung geht die Information über die Szene verloren*

- **Dateiname:** [p_DATASET_NAME]
Pflichtparameter zur Bezeichnung der aufbereiteten Dateien im Builder-Projekt. Der Dateiname muss je Szene individuell festgelegt werden, um das Überschreiben von bereits aufbereiteten Daten zu verhindern.



***Tipp:** Für die Parameter Szene und Dateiname lässt sich der optionale Parameter Kachel heranziehen. Durch Setzen von \$(p_Tile) bei Szene und Dateiname wird der Wert von Kachel übergeben. Derart bedarf es nur einer Änderung des Werts von Kachel um sicher zu stellen, dass Szene und Dateiname mit gleichen, eindeutigen Werten gesetzt sind.*

- **Log File:** [LOG_FILE]
Pflichtparameter um den Speicherort der FME-Protokolldatei fest-zulegen. Zu beachten ist, dass der Pfad zur Log-Datei bereits existieren muss, FME erzeugt keine fehlenden Unterverzeichnisse eines manuell gesetzten Log-Pfades.

III. CityGRID® 3D Studio Builder Plug-In

1 Installation und Start

Das 3D Studio Builder Plug-In ist technisch gesehen eine Erweiterung zum Funktionsumfang des Autodesk® 3D Studio Max, mit der 3D Daten arrangiert, Kamerafahrten erzeugt, sowie Animationen in einem Scout-Projekt erstellt werden können. Diese Daten werden in eine vorgegebene Ordnerstruktur gliedert und für die Optimierung des Scout-Projekts vorbereitet. Um das 3D Studio Builder Plug-In korrekt nutzen zu können, ist vor der Installation des CityGRID® Moduls eine unterstützte 3D Studio Version am Rechner zu installieren.



Hinweis: Die Liste der unterstützten 3D Studio Versionen kann im Handbuch CityGRID® Grundlagen nachgelesen werden. Während des CityGRID® Setups sind weiters alle unterstützten und am Rechner installierten 3D Studio Versionen wählbar.



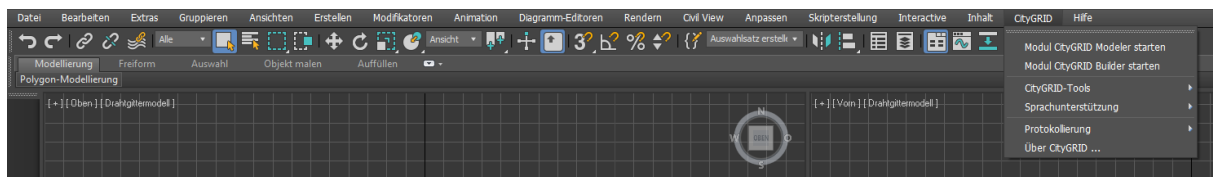
Hinweis: Das CityGRID® 3D Studio Builder Plug-In ist in der Basisversion von 3D Studio ebenso nutzbar, wie in der Design-Variante.

Beim jedem Versionswechsel von 3D Studio (etwa beim Umstieg auf eine neue Jahresversion) ist die gesamte Autodesk Komponente von CityGRID® zu deinstallieren und nach der Neuinstallation von 3D Studio wieder neu zu installieren.



Hinweis: Kontrollieren Sie in der Änderungsliste des ausgelieferten CityGRID® Setuppaketes ob die zu installierende 3D Studio Version auch durch CityGRID® unterstützt wird und kontaktieren Sie im Zweifel den CityGRID® Support.

Nach erfolgreicher Installation und Lizenzierung (vgl. Handbuch CityGRID® Grundlagen) erscheint in der Hauptmenüleiste von 3D Studio der Menüeintrag **CityGRID**, unter dem der Menüpunkt **Modul CityGRID Builder starten** zu finden ist.



1.1 Einstellen der Sprache

Die Sprache der Benutzeroberfläche steht derzeit auf Deutsch und Englisch zur Verfügung. Je nachdem, welche Sprache momentan aktiv ist, erscheint im Menü **CityGRID** der Menüpunkt **Sprachunterstützung**, oder **Localisation**. Wählen Sie darin die gewünschte Sprache der Benutzeroberfläche.

1.2 Starten des CityGRID® 3D Studio Builders

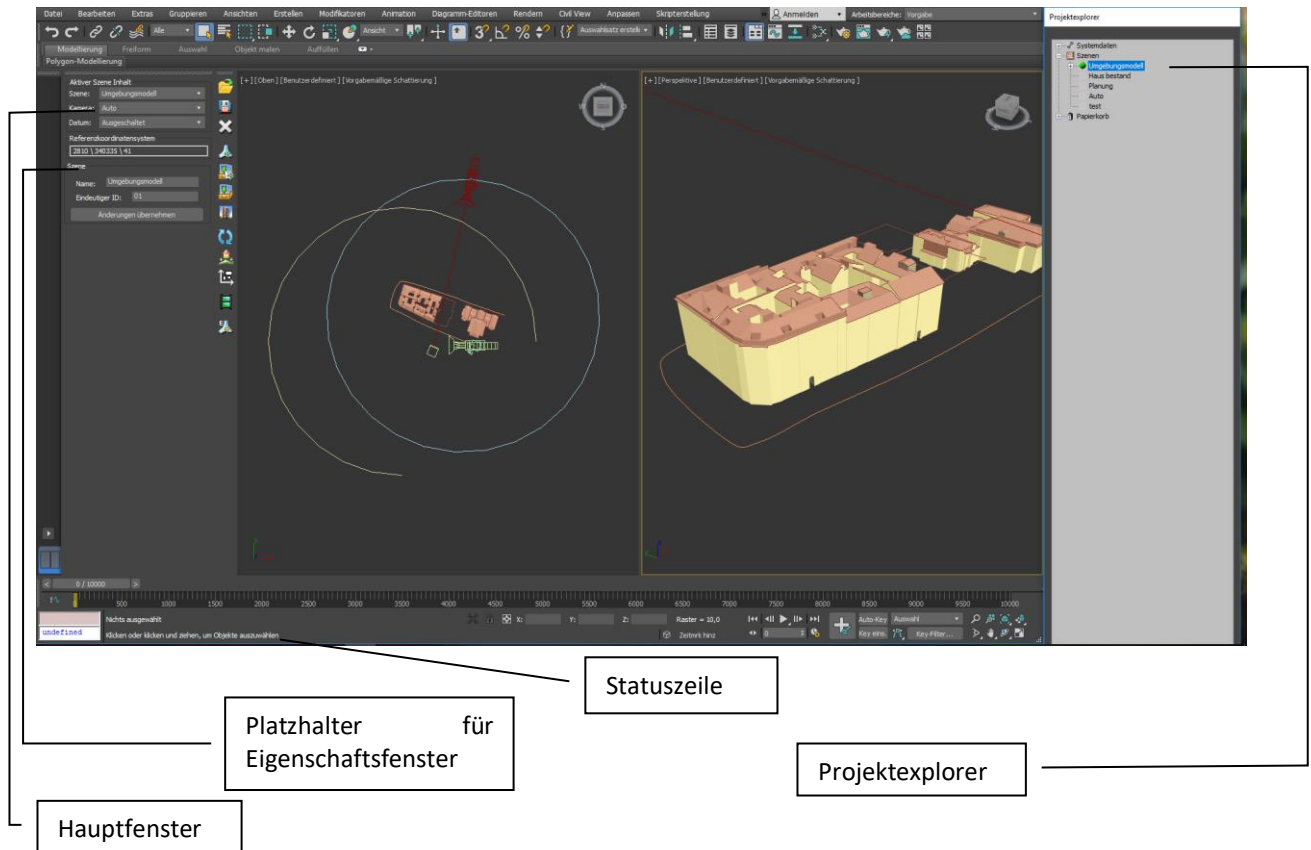
Sobald der Menüeintrag **Modul CityGRID Builder starten** geklickt wurde, wird die Oberfläche des 3D Studios angepasst und das Builder Plug-In gestartet. Unmittelbar nach dem Start erscheint sofort das Fenster *Projekteinstellungen* (vgl. III 3.1.1). In diesem ist entweder ein bestehendes Builder Projekt zu wählen oder ein neues Projekt anzulegen.



Hinweis: Das 3D Studio Builder Plug-In kann nur in Verbindung mit einem gültigen Builder-Projekt betrieben werden. Entweder muss ein neues Projekt angelegt, oder ein bereits bestehendes geladen werden. Wenn im Fenster *Projekteinstellungen* keine Auswahl getroffen wird, führt dies zur Beendigung des Builder Plug-Ins.

2 Die Benutzeroberfläche

Nach dem Starten des 3D Studio Builders wird die Benutzeroberfläche angepasst. Der Menüpunkt **CityGRID** wird ersetzt durch den Menüpunkt **CityGRID Builder**. Zusätzliche Fenster werden angedockt (das Hauptfenster des Builders und der Szene-Explorer) und eine Reihe zusätzlicher Buttons (die Haupt-Toolleiste) erscheinen.



Das Hauptfenster besteht aus den Schnellauswahl-Gruppen Projekt, Szene, Kamera und Datum. Zusätzlich sind die Offsetwerte des verwendeten lokalen Koordinatensystems angegeben (vgl. III 2.2).

Der Szene-Explorer enthält die Übersicht über den Aufbau des gesamten 3D Studio Builder-Projekts. Neben den Systemdaten (Sonnenstands- und Kameraeinstellungen) werden alle Szenen gelistet.













Funktionen werden entweder über die Haupt-Toolleiste oder über das Builder Menü gestartet. Optional erscheinende Eigenschaftsfenster oder Einstellungsfenster werden an den Szene-Explorer unten angedockt.

Wird eine Benutzeraktion erwartet, steht ein diesbezüglicher Hinweis in der Statuszeile.

Bei einem neuen Builder Projekt werden zwei Ansichtsfenster vorbereitet: links ein Grundrissfenster, rechts eine perspektivische 3D-Ansicht. Die Konfiguration der Fenster kann vom Benutzer angepasst werden und wird mit dem Projekt mit gespeichert.

Die Befehlspalette(Command-Panel) von Autodesk 3D Studio MAX wird standardmäßig ausgeblendet, da vorwiegend Funktionen des CityGRID® -Plugins zum Erstellen der Szene verwendet werden sollen. Bei Bedarf kann die Befehlspalette durch den Anwender jederzeit wieder sichtbar gemacht werden. Dazu ist in der Menüleiste des 3D Studios unter **Anpassen/UI anpassen** die Option **Befehlspalette anzeigen** zu wählen.

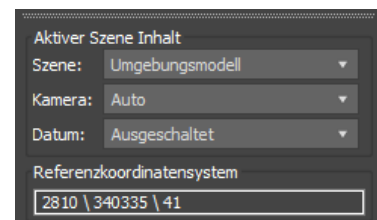
2.1 Buttons der Haupt-Toolleiste

	Neues Projekt erstellen / Projekt öffnen
	Projekt speichern
	Projekt beenden
	CityGRID® Daten übernehmen
	Bibliotheksobjekte hinzufügen
	Bibliotheksobjekte mittels einer externen Datendatei positionieren
	Externe Daten importieren
	Ansicht aktualisieren
	Objekt(e) auf das Gelände setzen
	Lage von Objekten ausrichten
	Animationen und Bilder rendern
	Source Daten für CityGRID® Scout berechnen

2.2 Hauptfenster

Das Hauptfenster enthält Auswahlmenüs um die momentane Ansicht in den 3D Studiofenstern definieren zu können. Konkret lassen sich folgende Einstellungen treffen:

- aktive Szene (vgl. III 3.2)
- aktive Kamera (vgl. III 5.1)
- Eingestelltes Datum für Sonnenstände (vgl. III 5.2)







Durch klicken auf das kleine Dreieck rechts öffnet sich ein Auswahlmenü für jede der genannten Auswahlen. Sobald ein Eintrag dieser Listen geklickt wird, passt sich die Ansicht im 3D Studio Max automatisch an.

Es werden weiters die Verschiebungswerte vom globalen Koordinatensystem in das lokale 3D Studio MAX Koordinatensystem angezeigt (vgl. III 3.1.1).


2.3 Projekt Explorer

Der Projekt Explorer enthält sämtliche Projektdaten in einer Baum-Darstellung und bildet die Gesamtheit aller Elemente des geladenen 3D Studio Builder-Projekts ab. Folgende Hauptknoten können sich im Projekt Explorer befinden:

- Systemdaten :
Darunter fallen projektbezogene Systemdaten, wie verschiedene Sonnenstands- und Kameraeinstellungen
- Szenen :
Ein Builder Projekt besteht zumeist aus mehreren Szenen, die in Kombination miteinander das gesamte Scout-Projekt bilden. Jede Szene sollte Daten einer bestimmten Datenklasse (z. B. Geländemodell, CityGRID® Gebäude, allgemeine 3D Daten) enthalten. Builder Projekte mit mehreren Szenen können im 3D Studio stets nur eine Szene aktiv geladen haben, was durch einen vorangestellten grünen Punkt  gekennzeichnet wird. Nur in dieser Szene können Veränderungen an den Daten vorgenommen werden.
- Papierkorb :
gelöschte Daten werden in den Papierkorb verschoben. Solange dieser nicht geleert wird, können Daten wieder restauriert werden.



Hinweis: Um den Inhalt des Papierkorbs endgültig zu löschen steht im Menü „CityGRID Builder“ der 3D Studio Max Hauptmenüleiste die Funktion Papierkorb löschen zur Verfügung. Durch Ausführen dieses Menüpunktes leert sich der Inhalt des Papierkorbs im Projektextplorer und die zugehörigen Backupdateien im Builder-Projekt werden gelöscht.

- Selektion :
Alle momentan im Viewport selektierten Objekte werden hier zusammengefasst und können auf einmal verschoben werden.

Durch Selektion eines Knotens im Projektextplorer wird das jeweilige Eigenschaften-Fenster (falls vorhanden) sowie die Exporteinstellungen für den CityGRID® Scout (vgl. III 8.1) an den Projektextplorer angedockt.

2.4 Drag & Drop

Knoten im Szene Explorer können mittels Drag & Drop verschoben und kopiert werden. Davon ausgenommen sind die Hauptknoten Systemdaten, Projekte, Varianten und Papierkorb. Weiters sind der momentan aktive Sonnenstand, die aktive Kamera sowie das geladene Projekt für Verschiebungen generell gesperrt. Alle übrigen Verschiebungen sind jederzeit möglich und werden im Projektextplorer sofort angezeigt.

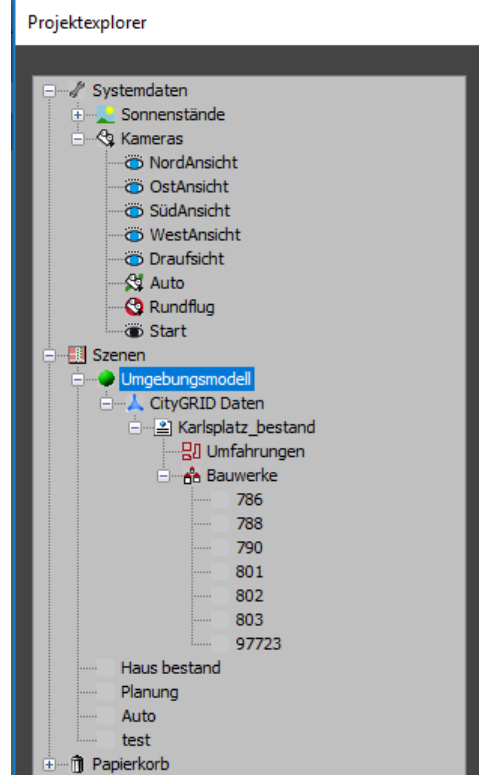
Klicken Sie mit der linken Maustaste auf den Knoten im Projektextplorer, der die zu verschiebenden Daten repräsentiert.

Halten Sie die Maustaste gedrückt und ziehen Sie die Maus über den Knoten im Projektextplorer, zu dem die Daten verschoben werden sollen. Sobald ein Knoten erreicht ist, zu dem die Daten verschoben werden dürfen, ändert sich der Mauszeiger.

Wenn der Zielknoten im Projektextplorer erreicht ist, lassen Sie die Maustaste los.



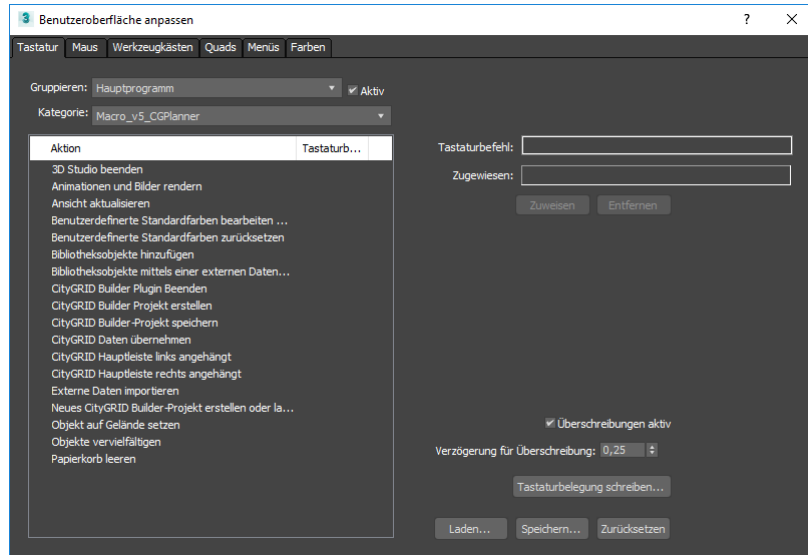
Tip: Um Mehrfachselektionen per Drag & Drop verschiebbar zu machen, gibt es im Projektextplorer den Eintrag Selektionen. Er wirkt auf alle selektierbaren Objekte des Builder-Projekts und erscheint sobald ein Objekt im Viewport, oder ein Blatt (die hierarchisch unterste Ebene im Projektextplorer) im Projektextplorer angeklickt wird. Mehrfachselektionen sind nur über eine Auswahl im Viewport möglich.



Das Verschieben des Selektionen Knoten im Projektextplorer wirkt auf alle selektierten Objekte gleichzeitig.

2.5 Festlegen von Tastenschlüsseln (Hotkeys) in 3D Studio MAX

Eine Reihe von Aktionen des CityGRID® Builders lassen sich zusätzlich zu Klicks auf entsprechende Buttons über vom Benutzer zu definierende Tastaturbefehle aufrufen. Die Zuweisung solcher „hotkeys“ erfolgt durch Öffnen des Menüs „Anpassen – Benutzeroberfläche anpassen“ („Customize – Customize User Interface...“). Auf dem Reiter *Tastatur (Keyboard)* kann für die Gruppe *Hauptprogramm (Main UI)* die Kategorie *Macro_v5_CGPlanner* ausgewählt werden.



Jeder der nachfolgend dargestellten Aktionen des CityGRID® Builders kann ein Hotkey zugewiesen werden. *Speichern (Save)* übernimmt die definierten Hotkeys, sowie die momentane Konfiguration der Arbeitsoberfläche. Die Position von Toolbars etwa kann so individuell festgelegt werden.

2.6 Ändern der Farben von CityGRID® Daten

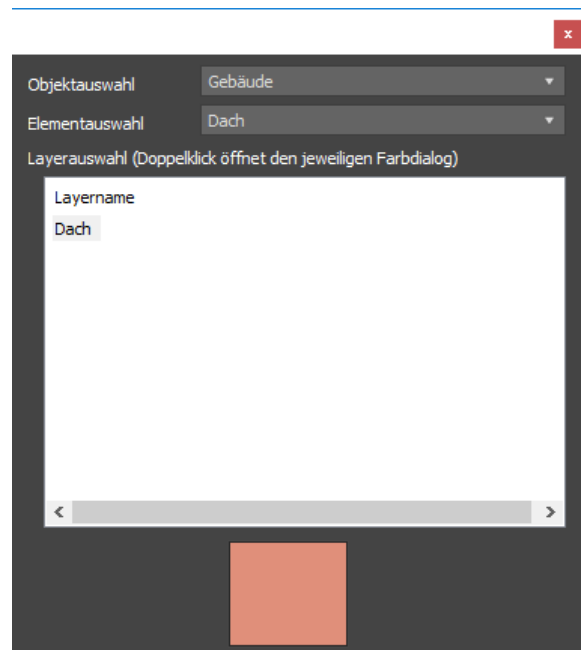
Die Standardfarben für Flächenlayer von CityGRID® Daten können angepasst werden.



Hinweis: Änderungen, die im CityGRID® 3D Studio Builder vorgenommen werden, werden für diesen Benutzer auch im CityGRID® Modeler wirksam und umgekehrt!




Die benutzerdefinierten Standardfarben für Flächenlayer von Units können über das Menü **CityGRID Builder> Benutzerdefinierte Standardfarben...** eingestellt werden. Es öffnet sich ein Dialog, in dem für jede Objektklasse und Unterelementklasse und für den Flächenlayer eine Farbe eingestellt werden kann.

Der Farbauswahl-Dialog kann durch Doppelklicken auf den jeweiligen Layernamen oder durch Klicken auf das Farbfeld geöffnet werden.



3 Arbeiten mit dem CityGRID® 3D Studio Builder

Die Hauptfunktion des CityGRID® Builder ist das Zusammenstellen von Visualisierungen mit Daten aus verschiedenen Quellen sowie das Aufbereiten verschiedener Planungsvarianten. Geometriedaten können aus den folgenden Quellen stammen:

- CityGRID® Daten 
3D-Modelle von Gebäuden und anderen 3D Objekten (Mauern, Brücken, etc.), die in einer CityGRID® Datenbank verwaltet, oder in CityGRID® XML Dateien vorgehalten werden, können in das Projekt aufgenommen werden (vgl. III 3.3).
- Projektdaten 
3D-Modelle von geplanten Gebäuden und anderen 3D Objekten, die entweder aus externen Dateien importiert, oder direkt mittels 3D-Studio Operationen erstellt werden. CityGRID® Builder-Projekte werden ebenfalls als Projektdaten in den CityGRID® Builder übernommen (vgl. III 4.2).
- Bibliotheksdaten 
Parametrisierte 3D-Modelle, die aus Bibliotheksmodellen erstellt werden (mit Parametern wie Breite, Länge, Höhe, etc.; vgl. III 4.3)

3.1 Projekte

Ein CityGRID® Builder-Projekt besteht im Dateisystem aus einer Serie von Unterverzeichnissen. (vgl. II 2). Eines dieser Unterverzeichnisse trägt die Bezeichnung „Builder“. In diesem finden sich alle Szenen eines 3D Studio Builder-Projekts.

__Deleted__: Hier werden zu löschende Daten, die im 3D Studio Builder auf den Papierkorb verschoben wurden, zwischengespeichert.

__Media__: Hier entstehen Bilder und Videos, die im 3D Studio Builder gerendert werden.

__Projects__: Hier werden alle Szenen verspeichert.


__Textures__: Hier werden Texturen verspeichert, die im 3D Studio Builder angelegt werden (z.B. das Texturbild für den Geländeausschnitt)

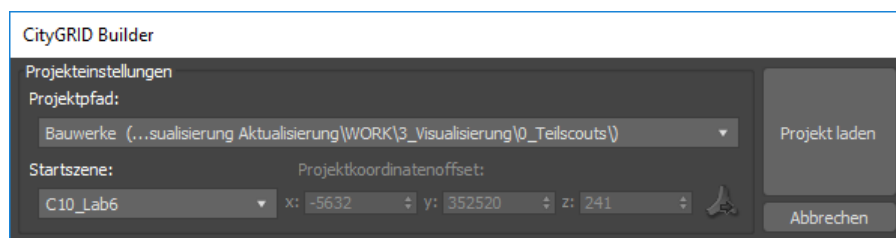


***Hinweis:** Die gelisteten Verzeichnisse bilden nur die Daten des 3D Studio Builder ab und repräsentieren kein vollständiges Builder-Projekt. Dieses ist nur in der unter 2 beschriebenen Verzeichnisstruktur gültig.*

Im Supportfall senden Sie, nach Aufforderung, bitte stets das gesamte Builder-Projekt an UVM Systems!

3.1.1 Neues Projekt erzeugen/ Projekt laden

Das Fenster zum Erstellen eines neuen CityGRID® Builder-Projekts wird automatisch nach dem Start des Moduls geöffnet. Es kann aber auch später über den Button Neues Projekt  der Haupt-Toolleiste geöffnet werden.



1. Projektpfad:

In der Auswahlliste kann ein bereits bestehendes Builder-Projekt ausgewählt werden, oder durch Klick auf *select new* ein neues Projekt angelegt werden.



***Tipp:** Bereits vorhandene Projekte werden sowohl mit dem Projektnamen als auch mit dem Speicherort aufgeführt.*

Wenn ein bereits vorhandenes Projekt gewählt wurde, werden die übrigen Parameter des Fensters *Projekteinstellungen* automatisch vorgenommen.

Beim Anlegen eines neuen Projekts öffnet sich ein Filebrowser in dem man den Speicherort des Builder-Projekts angeben muss. Über Standard-Windowsfunktionen können im Browser neue Verzeichnisse angelegt werden. Sobald ein Verzeichnis gewählt wurde, wird in diesem die Builder-Verzeichnisstruktur (vgl. III 2) angelegt und das Projekt dadurch eingerichtet.



***Tipp:** Builder-Projekte können teilweise riesige Datenmengen enthalten. Bei der Definition des Speicherorts ist daher auf ausreichend Platz zu achten. Während der Bearbeitung, insbesondere während der Scout-Erstellung, empfiehlt es sich weiters auf einer lokalen Festplatte zu arbeiten, da in diesem Arbeitsschritt sehr viele Dateien entstehen die auf Festplattenspeicher zwischengelagert werden. Beim Arbeiten über ein Netzwerk führt dies zu unnötigem Traffic und Performanceeinbußen.*

2. Startprojekt:


Falls das Builder-Projekt aus mehreren Szenen besteht, kann über diese Auswahlliste festgelegt werden, welche Szene ins 3D Studio geladen werden soll. Alle übrigen Szenen werden als Einträge in den Projektextplorer aufgenommen und können während des Arbeitens mit dem 3D Studio Builder gewechselt werden. Bei einem neu angelegten Builder-Projekt steht die Auswahl des Startprojekts nicht zur Verfügung da noch keine Daten vorhanden sind.

3. Projektkoordinatenoffset:

Die Komposition der Szenen im 3D Studio Builder erfolgt in einem lokalen Projektkoordinatensystem. Da 3D Studio keine ausreichende Zahlenschärfe bietet, verschiebt CityGRID® für die Bearbeitung die geladenen Koordinaten von CityGRID® Gebäuden selbständig in ein lokales Koordinatensystem mit reduzierten Koordinaten. Sobald die Daten prozessiert und im Source Ordner abgelegt werden, wird der Projektkoordinatenoffset wieder angebracht, sodass die Daten wieder im Ausgangskordinatensystem vorliegen.

Der Projektkoordinatenoffset wird global für das ganze Builder-Projekt angegeben und gilt für jede unter 3D Studio erzeugte Szene. Ein nachträgliches Ändern des Offsets ist nicht möglich.

Wahlweise kann durch den Nutzer ein Offsetwert angegeben werden, indem Verschiebungswerte für x, y und z angegeben werden. Die angegebenen Werte werden von den Ausgangskordinaten abgezogen

um zu den lokalen Projektkoordinaten zu gelangen. Optional kann über den Button  Projektkoordinatenoffset von CityGRID Daten übernehmen auch ein CityGRID® Modell ausgewählt werden, von dem dann automatisch die Reduktionswerte ermittelt werden.



***Hinweis:** Um spätere Passungenauigkeiten beim Zusammenspiel von CityGRID® Gebäuden mit externen Daten (z. B. Architekturmodellen) zu vermeiden, sollten sämtliche externe Daten vor einem Import in ein einheitliches Lagebezugssystem transformiert, oder an den Projektkoordinatenoffset des Builder-Projekts angeglichen werden.*

4. Projekt öffnen/erstellen:

Projekt laden


Durch klicken auf den Button wird das Builder-Projekt geladen. Falls ein neues Builder-Projekt begründet wurde, werden folgende Aktionen ausgeführt:

- Im Dateisystem wird ein neues Builder-Projekt mit der vordefinierten Ordnerstruktur angelegt.
- Die Ansichtsfenster werden standardmäßig zurückgesetzt (ein Grundriss- und ein 3D-Fenster).
- Eine Szene mit dem Namen „Standard“ wird generiert.
- Fünf standardmäßig vorbereitete Kameraeinstellungen vom Typ „Relative Kamera“ (vgl. III 5.1) werden generiert: eine Nord-, Ost-, Süd- und Westansicht sowie eine Draufsicht.
- Ebenso werden acht standardmäßig vorbereitete Sonnenstände (vgl. III 5.2) generiert: zu den 4 Jahreszeiten-Wechsel jeweils mit fixem Zeitpunkt mittags und als Zeitspanne von morgens bis abends. Dabei wird standardmäßig der zum Installationszeitpunkt angegebene Ort verwendet.

Anschließend kann der Benutzer Daten hinzufügen (vgl. III 4)

3.1.2 Projekt speichern

Das Projekt wird vielfach automatisch zwischengespeichert. Insbesondere vor dem Wechsel zu einer anderen Szene, und damit dem Entladen der aktiv geladenen Szene aus 3D Studio, wird die aktuelle Szene gespeichert. (vgl. III 3.2.2)

Zusätzlich sollte regelmäßig durch Klicken auf den Button **Speichern**  der Haupt-Toolleiste das Projekt gesichert werden. Dabei werden sämtliche Daten sowie die Ansichtseinstellungen gespeichert.



***Tipp:** Um lange Speicherzeiten zu vermeiden sollte ein Builder-Projekt stets lokal gespeichert werden. Der Betrieb von Projekten auf Netzlaufwerken ist möglich aber aus Effizienzgründen mitunter nicht zu empfehlen!*

3.2 Szenen

Jedes Builder-Projekt besteht aus zumindest einer, häufig jedoch mehreren Szenen. Da Builder-Projekte sehr oft große, und detaillierte Kompositionen aus unterschiedlichen 3D Daten sind, ist es auf Grund der Handhabbarkeit und Übersichtlichkeit, sowie der einfacheren Aktualisierbarkeit angeraten das Projekt auf mehrere Teilbereiche, die Szenen, zu unterteilen. Jede Szene enthält einen bestimmten Teil des gesamten Builder-Projekts und die Summe aller Szenen setzt das Builder-Projekt zusammen. Die einzelnen Szenen können sowohl über den 3D Studio Builder, als auch über den FME Builder angelegt werden.



***Hinweis:** Falls das Projekt sehr klein ist, kann gegebenenfalls auch das ganze Projekt in einer Szene gekapselt werden.*

Im 3DStudio können beliebig viele Szenen angelegt, jedoch stets nur eine Szene aktiv in den Viewport geladen werden. Alle übrigen Szenen sind im entsprechenden Verzeichnis des Builder-Projekts abgelegt und aus 3D Studio entladen. Durch Auswahl der gewünschten Szene im Projektexplorer kann zwischen vorhandenen Szenen gewechselt werden. Dabei wird die aktive Szene aus 3D Studio entladen und die gewählte Szene importiert.

3.2.1 Neue Szene anlegen

Szenen werden, wie jede andere neue Komponente im 3D Studio Builder auch, durch selektieren des Hauptknotens im Projektexplorer angelegt. Sobald der Knoten Szene geklickt wurde, öffnet sich ein Eigenschaftsfenster in dem der Szenenname, sowie ein eindeutiger Projektcode einzugeben ist.

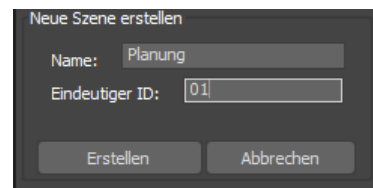
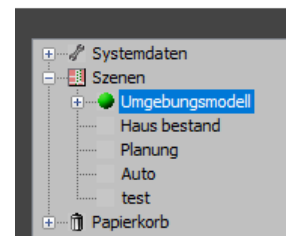
- **Namen**

Geben Sie der neuen Szene einen eindeutigen *Namen*

- **Eindeutiger ID**


Eindeutiger ID dient dazu der Szene einen eindeutigen Bezeichner, neben dem Szenenamen, zu übergeben. Dieser Parameter wird für die Verwaltung der Szene verwendet. Der Szene ID darf maximal 8 Zeichen lang sein und sowohl Zeichen als auch Ziffern enthalten. Falls eine bereits vergebene ID nochmals gesetzt wird, erscheint eine entsprechende Fehlermeldung.

Projektexplorer




***Hinweis:** Der Szene ID bietet für den Nutzer auch die Möglichkeit der Projektadministration. Durch Einführung eines Bezeichnungsschemas lässt sich bei großen Scout-Projekten so rasch ein Überblick über unterschiedliche Szenen herstellen. Beispielsweise könnte ein Bezeichnungsschema des ersten Zeichens folgender Logik folgen: 1 = Geländemodell (StaticTerrain), 2 = Gebäudemodell (StaticBuilding), 3 = Allgemeine 3D Daten (StaticDesign), 4 = Punktwolken (StaticPoint), 5 = Instanzobjekte (StaticInstance). Die Szene ID ist für das gesamte Builder-Projekt gültig und kann daher von 3D Studio als auch von FME für jede Szene gesetzt werden.*

- *Erstellen*

Mittels  wird die neue Szene angelegt. Anschließend wird die Szene im Projektextplorer unter dem Knoten Szene hinzugefügt und aktiviert. In weiterer Folge können nun Daten in diese Szene eingefügt werden.

3.2.2 Wechseln der aktiven Szene

In einem CityGRID® Builder Projekt ist immer eine Szene im Viewport aktiv, und zeigt die zugehörigen Daten im Viewport von 3D Studio an. Nicht aktive Szenen werden von 3D Studio entladen um Speicherplatz freizugeben und befinden sich im Dateisystem des Builder-Projekts.

Die geladene Szene wird im Projektextplorer durch einen grünen Punkt  gekennzeichnet. Um eine Szene zu wechseln kann entweder im Hauptfenster eine andere Szene im Pulldown Menü gewählt werden, oder im Projektextplorer die gewünschte Szene direkt angeklickt werden.

Da andere Szenen entladen sind, kann der Inhalt einer Szene nur für die aktive ausgeklappt werden.



***Hinweis** Beim Wechsel der aktiven Szene wird das CityGRID® 3D Studio Builder-Projekt automatisch gespeichert, da die zuletzt aktive Szene entladen wird.*

3.3 Anforderungen an 3D Objekte für den CityGRID® Scout

3D Objekte müssen nachstehende Bedingungen erfüllen, um ordnungsgemäß im CityGRID® Scout angezeigt zu werden.

Alle CityGRID® Geometrieobjekte erfüllen nachstehende Bedingungen standardmäßig, auf die Einhaltung ist bei der individuellen Erstellung von Bibliotheksobjekten (vgl. III 4.3), sowie der Integration von externen Modellen (vgl. III 4.2) zu achten.

3.3.1 Geometrie

Es können Geometrie-Objekte eingesetzt werden, die vom 3D Studio Typen „Mesh“, „Trimesh“ oder „EditableMesh“ sind, bzw. all jene Geometrie-Objekte, die in die zuvor genannten drei Mesh-Typen konvertierbar sind. Zu beachten ist, dass es bei der Konversion, beispielsweise von Freiformflächen (Patches), durch 3D Studio zu Geometrieveränderungen vorkommen kann.

Zur effizienten Flächendarstellung verwendet CityGRID® Scout ausschließlich die Flächennormalen der 3D Flächen. Eine beidseitige Darstellung ist nicht vorgesehen! Optionale Darstellungsvarianten in 3D Studio werden beim Export in CityGRID® Scout ignoriert (z.B. beidseitige Flächendarstellung, Materialtyp einseitige Flächendarstellung etc.)

3.3.2 Material


Es werden nur Standardmaterial und Multimaterial (bestehend aus Standardmaterialien) unterstützt. Aus dem Standardmaterial werden nur die „Streufarbe“ und die „Streufarben Map“ ausgewertet, alle übrigen „Maps“ werden ignoriert. Beispielsweise können Architekturmaterialien nicht übernommen werden. In der „Streufarben Map“ wird nur der Map-Typ „Bitmap“ bzw. „Bitmap 2D“ unterstützt. *Cropping* und *Placement* für Bitmap dürfen im Material Editor nicht verwendet werden! Eine Anwendung von *Cropping/Placement* bewirkt ein Fehlverhalten des 3D Objekts im CityGRID® Scout.

4 Daten in ein CityGRID® Builder-Projekt einbinden

4.1 CityGRID® Daten zu einem Builder-Projekt hinzufügen

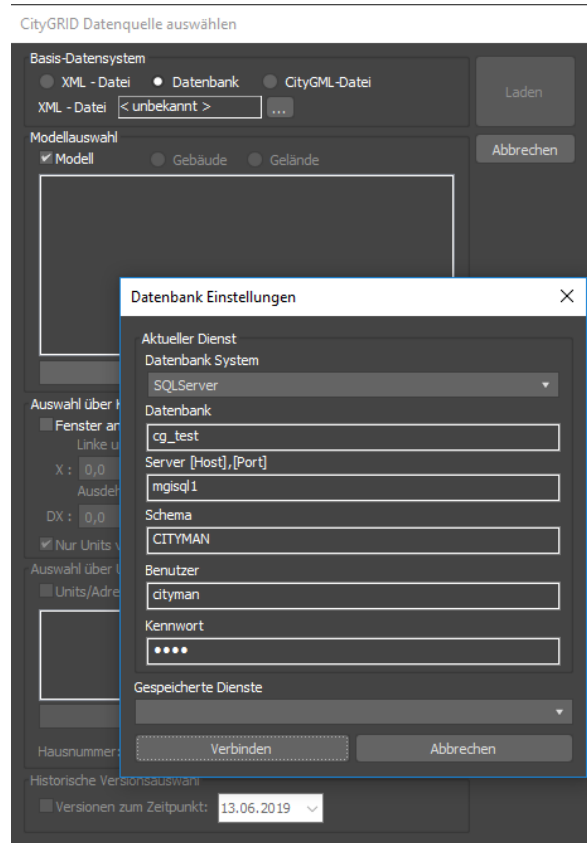
Am Beginn steht die Definition des Arbeitsgebiets und der Datenquelle. Dies erfolgt im Auswahlfenster.

4.1.1 Auswahlfenster für Datenbankverbindungen öffnen

1. Über den Button **CityGRID Daten importieren**  der Haupt-Toolleiste wird das Auswahlfenster geöffnet.
2. In den Datenbank-Modus wechseln: Auswahl **Datenbank** in der Selektion **Basis Datensystem**
3. Datenbankeinstellungen-Fenster über den Button **DB** öffnen.
4. Name des installierten Datenbank-Dienstes, Name des Datenbank-Schemas der CityGRID-Tabellen, Benutzernamen und Kennwort eingeben (Diese Daten vom Datenbank-Administrator erfragen).




***Hinweis:** Gültige Einstellungen werden gespeichert und können später einfach über die Selektion „Gespeicherte Dienste“ geladen werden. Durch das Speichern des Server- und Datenbanknamens /Dienstes können auf einem MSSQL Server auch mehrerer Datenbanken rasch aufgerufen werden. Beim Starten des Modelers wird immer versucht, die zuletzt aktive Datenbankverbindung wiederherzustellen.*



4.1.2 Auswahlfenster für XML Dateien öffnen

CityGRID® Daten können anstatt aus der Datenbank auch aus einer Datei im vom UVM Systems definierten XML-Format geladen werden:

1. Über den Button **CityGRID Daten importieren**  der Haupt-Toolleiste wird das Auswahlfenster geöffnet.
2. In den XML-Datei-Modus wechseln: Auswahl **XML-Datei** in der Selektion **Basis Datensystem**
3. Über den Button **...** die XML-Datei laden.

4.1.3 Arbeitsbereich definieren

Für das effiziente Arbeiten sollten Units zu Modellen zusammengefasst werden (siehe Handbuch CityGRID® Grundlagen). Zum Laden von Modellen muss in der Rubrik **Modellauswahl** auf **Modell** gestellt werden.

Optional können neben Modellen auch Geländemodelle im Auswahlfenster zum Laden angezeigt werden. Zum Laden von Geländemodellen muss in der Rubrik **Modellauswahl** auf **Gelände** gestellt werden.

Zur Auswahl der zu ladenden Daten stehen folgende Methoden zur Verfügung:

1. **Modellauswahl:**

Checkbox *Modell* aktivieren und eines der Modelle auswählen.

2. **Auswahl über Koordinaten:**

Checkbox *Fenster angeben* aktivieren und die linke untere Ecke sowie die Ausdehnung angeben. Optional kann die Checkbox *Nur Units verwenden, die vollständig innerhalb des Bereichs liegen* gesetzt werden um angeschnittene Units von der Selektion auszuschließen.

3. **Auswahl über UnitID / Adresse:**

Checkbox *Units/Adresse auswählen* aktivieren und im Fenster Unitauswahl die Suchkriterien eingeben. Über das Pull Down Menü kann zwischen *UnitID* und *Straße* ausgewählt werden.

○ **UnitID:**

durchsucht die Datenbank nach den angegebenen Namen, der links neben dem Pull Down Menü eingegeben wird. Um mehrere Units zu laden kann ein beliebiger gemeinsamer Teil des UnitIDs (in korrekter Groß- / Kleinschreibung) eingetippt werden, Platzhalter für fehlende Zeichen werden nicht benötigt.



Tipp: Über die Unitauswahl ist es möglich einzelne Units oder auch Unitgruppen zu laden, ohne ein Modell, in dem diese Units liegen, laden zu müssen. Für eine sinnvolle Nutzung der Unitauswahl ist die Vergabe von strukturierten UnitIDs dringend angeraten!

○ **Straße:**

lädt alle Units die einem Straßennamen zugeordnet werden können. Zum Einschränken der Auswahlmöglichkeiten kann man einen beliebigen Teil des Straßennamens (in korrekter Groß- / Kleinschreibung) in das Textfeld links neben dem Pull Down Menü eingeben.

Zur weiteren Einschränkung auf *Hausnummern* kann im gleichnamigen Feld eine vorkommende Nummer oder ein Bereich angegeben werden.

Der Parameter *Bereich vergrößern* legt um die gefundenen Units einen Puffer in der angegebenen Größe und lädt alle darin befindlichen, oder schneidend liegenden Units.



Hinweis: Die Unitauswahl ist nur im Datenbank-Modus möglich, und die Straßenauswahl auch nur dann, wenn eine Verknüpfung zu einer Adressdatenbank installiert ist. Um eine derartige Verknüpfung zu installieren, wenden Sie sich bitte an UVM Systems.

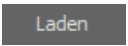


Hinweis: Das Abfragen aller UnitIDs in der Datenbank kann sehr zeitaufwendig werden. Daher sollte die Auswahl über UnitIDs nur im Bedarfsfall durchgeführt werden

Falls mehrere Auswahl-Möglichkeiten kombiniert werden, wird die gemeinsame Menge der Units geladen.



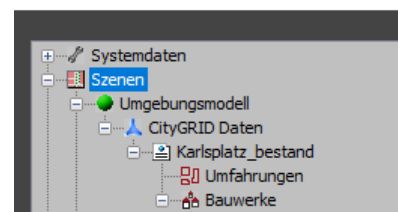
Beispiel: Mit Modellauswahl „Innenstadt“ und Straßenauswahl „Hauptstraße“ kann jener Teil der Hauptstraße zum Arbeitsgebiet definiert werden, der innerhalb der Innenstadt liegt.

Mit  wird das Auswahlfenster geschlossen und im Projektextplorer unter der aktiven Szene ein Knoten CityGRID Daten eingefügt. Darunter befindet sich ein weiterer Eintrag, mit dem soeben geladenen Modell. Die Auswahlobjekte aller Units im Selektionsgebiet werden geladen, und diese Auswahlobjekte werden im Projektextplorer als Knoten *Umfahrung* eingefügt.



Hinweis: Die Auswahlpolygone zur Auswahl der zu ladenden Units, werden im Ansichtsfenster nur dann aufgebaut, wenn im Projektextplorer der Knoten „Umfahrungen“ selektiert ist. Bei umfangreichen Modellen kann dies zu kurzen Ladezeiten führen.

Projektextplorer



4.1.4 Laden der CityGRID® Daten

Über das Eigenschaftsfenster *CityGRID Daten übernehmen*, das unterhalb des Projektextplorers erscheint, sobald die *Umfahrungen* im Projektextplorer angeklickt werden, erfolgt die Auswahl der zu ladenden CityGRID® Daten. Dabei wird zwischen Bauwerken und Geländemodellen unterschieden.

Um Bauwerke zu laden ist folgendermaßen vorzugehen:

1. Die Checkbox bei *Bauwerke (neu) übernehmen* setzen.
2. Wenn mit einer CityGRID® Datenbank gearbeitet wird, kann über den *LoD (Level of Detail)* die geometrische Ausprägung des Bauwerks festgelegt werden (vgl. Handbuch CityGRID® Grundlagen)



Hinweis: Die Auswahlobjekte werden in unterschiedlichen Farben je nach dem Status der letzten gültigen Version dargestellt. Sofern die Farben nicht verändert worden sind, bedeutet:

Hellgrau: Die letzte gültige Version ist eine stabile, nicht abgerissene Version.

Rot: Die letzte gültige Version wurde im CityGRID® Modeler ausgecheckt (vermutlich arbeitet ein anderer Benutzer daran). Lädt man diese Unit, wird die letzte stabile Version geladen.

3. Falls die Bauwerke Texturen haben, und diese auch verwendet werden sollen, muss die Checkbox bei *Texturen übernehmen* gesetzt werden.
4. Wenn mit einer CityGRID® Datenbank gearbeitet wird, kann über die *Texturauflösung* festgelegt werden, wie hochauflösend die geladenen Bilder des Bauwerkes sein sollen. Der angegebene Wert im Pulldown-Menü gibt dabei das Verhältnis von einem Pixel zur realen Größe des Objekts in Millimetern an. XML Dateien verwenden automatisch immer die höchstmögliche Auflösung.
5. Zur weiteren Einschränkung der zu ladenden Bauwerke kann die Option *Nur ausgewählte verwenden* gesetzt werden. Bei dieser Methode erfolgt eine Untersuchung auf im Viewport selektierte Auswahlpolygone. Bei Option *Alle Auswahlpolygone verwenden* werden sämtliche Bauwerke des geladenen Modells aus den Auswahlpolygonen aufgebaut.



Hinweis:

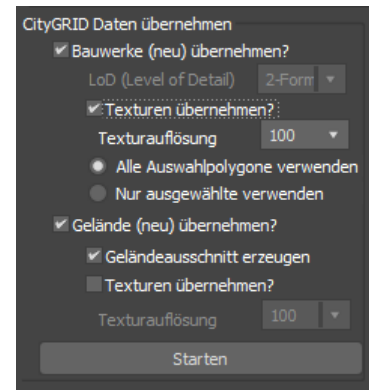
Die Units werden in der CityGRID® Datenbank in verschiedenen Detaillierungsstufen gespeichert und können in jeder dieser Stufen geladen werden.

Folgende LoD-Stufen sind implementiert:

- 1 ... Klötzchenmodell
- 2 ... Formmodell (mit Dachform und ev. Dachvorsprung)
- 3 ... Detailmodell (incl. Gauben etc.).

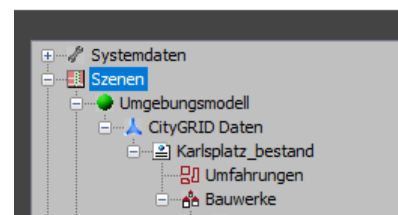
Um Geländemodelle zu laden, ist folgendermaßen vorzugehen:

1. Die Checkbox *Gelände (neu) übernehmen* setzen.
2. Die Option *Geländeausschnitt erzeugen* lädt nur jene Flächen von Geländemodellen die schneidend zu den geladenen Bauwerken, oder den selektierten Umfahrungspolygonen liegen. Wenn die Option nicht gesetzt ist, werden alle Geländedreiecke aufgebaut.
3. Falls die Geländemodelle Texturen haben, und diese auch verwendet werden sollen, muss die Checkbox bei *Texturen übernehmen* gesetzt werden.
4. Wenn mit einer CityGRID® Datenbank gearbeitet wird, kann über die *Texturauflösung* festgelegt werden, wie hochauflösend die geladenen Bilder des Geländes sein sollen. Der angegebene Wert im Pulldown-Menü gibt dabei die Größe der Bodenpixel an. XML Dateien verwenden automatisch immer die höchstmögliche Auflösung.



Durch Klicken auf **Starten** werden Kopien der CityGRID® Daten im Builder-Projekt erzeugt. Dieser Vorgang kann bei größeren Gebieten einige Minuten dauern. Während dieser Zeit erscheint ein Status-Fenster. Schließlich wird im Projektexplorer ein Knoten *Bauwerke* unter dem Knoten des geladenen CityGRID® Modells angelegt, unter dem die einzelnen Units mit ihrer UnitID zu finden sind. Optional kann auch ein Knoten *Gelände* gebildet werden.

Projektexplorer



4.2 Fremddaten zu Builder-Projekt hinzufügen

Fremddaten sind im alle Daten, die außerhalb der aktuellen CityGRID® Softwaresuite generiert wurden. Darunter fallen etwa externe 3D-Modelle von Architekten oder Planern, aber auch alle anderen im 3D-Studio erzeugten Modelle.


4.2.1 Anforderungen an Fremddaten

Für den optimalen Einsatz von Daten, die außerhalb der CityGRID® Software erstellt wurden sind eine Reihe von Vorgaben zu berücksichtigen, damit das Einpflegen der Daten in das Builder-Projekt problemlos funktioniert.

- Die Geometrie ist in Form einer Dreiecksvermaschung (Mesh) zu erstellen, NURBS Oberflächen sind nicht zulässig.
- Die Gebäude sollten korrekt in 3D platziert sein. Eine nachträgliche Georeferenzierung ist zumeist extrem aufwändig und nur bedingt genau. Der Einsatz eines pauschalen Offsetwerts um die geforderte Koordinatenschärfe erhalten zu können, ist aber zulässig. Sollte mehr als ein Gebäude in unterschiedlichen Dateien modelliert werden, ist immer derselbe Offsetwert zu verwenden. Die Werte des Offsets sind im Dateinamen zu führen.
- Die Flächennormalen sind ausnahmslos so zu orientieren, dass diese stets vom 3D Objekt weg zeigen. Dies entspricht eine Digitalisiererrichtung gegen den Uhrzeigersinn, auf die bereits bei der Modellerstellung zu achten ist
- Die Zeicheneinheit muss der Einheit der übrigen dargestellten Daten entsprechen. Zumeist wird die Einheit Meter sein.
- Die Geometrie soll die Charakteristik des Gebäudes wiedergeben, aber möglichst sparsam mit den Flächen umgehen. Der Grundsatz lautet: So wenig Flächen wie möglich, so viele wie nötig.
- Gebäudedetails, die durch Textur dargestellt werden können, sind nicht aus zu modellieren. Insbesondere Zäune, Balkongeländer, Gitter aller Art müssen mittels Textur gemacht werden.
- Innenräume der Gebäude können komplett ausgespart werden. Das Navigieren im Gebäudeinneren (etwa in Wohnungen) ist mit dem Scout nicht ideal und sollte daher vermieden werden. Große Hallen hingegen können auch von innen beflogen werden, wenn die entsprechende Innenausstattung modelliert wurde. Insbesondere Wände müssen für die Innenansicht mit den Flächennormalen in Richtung Innenraum erneut erstellt werden.
- Die Geometrie muss überall dort Dreieckskanten aufweisen, wo unterschiedliche Materialien aufgebracht werden sollen. Jede Dreiecksfläche kann nun genau eine Materialinformation aufnehmen.
- Neben der Gebäudegeometrie muss auch die Geometrie des Geländemodells (Planungsmodell) im unmittelbaren Nahbereich erstellt werden. Die geplanten Gebäudemodelle müssen exakt auf das Planungsmodell aufgesetzt sein, damit hochwertige Visualisierungen entstehen und insbesondere ein Navigieren aus der Fußgängerperspektive ermöglicht wird.
- Das Material muss vom Typ „Standard“ (in 3D Studio Max) sein. Etwaige Spezialmaterialien, die ein realitätsnahes Oberflächenverhalten (Glanz, Oberflächenbeschaffenheit, etc.) erlauben, können in der Echtzeitvisualisierung nicht verwendet werden, da die zu berechnenden Effekte mit der vorhandenen Hardware nicht schnell genug zu rendern sind. Für eine flüssige Bewegung sind 25, besser 50 Frames pro Sekunde notwendig.

- Vom Standardmaterial wertet der Builder nur die Streufarbe (Diffuse Color) und den Opazitätskanal aus. Somit dürfen auch nur diese Werte definiert sein, alle anderen Werte werden bei der Übergabe an den Scout ignoriert.
- Wenn ein Texturbild bei der Streufarbe verwendet wird, muss dieses vom Typ „Bitmap“ sein und vom Typ .jpg, .png oder .tif sein. Wenn ein Alphakanal zu nutzen ist, sind stets .png Bilder zu verwenden. Der Scout kann den Alphakanal direkt aus dem png nehmen und anwenden.
- Flächen die mit Alphakanal-Texturbildern belegt sind müssen im Modell geometrisch doppelt, mit entgegengesetzter Normalenausrichtung vorkommen. Andernfalls wird das Flächenobjekt nur von einer Seite aus sichtbar sein. Es ist nicht ausreichend das Material auf doppelseitige Darstellung zu setzen, diese Information wird vom Scout nicht ausgewertet! Alternativ zur Verdoppelung der Flächen ist es auch zulässig ganz dünne 3D Körper (z.B. Quader) zu bilden.
- Bei jedem Material das Texturbilder nutzt, muss der Bildmaßstab gleichbleiben. Dies wird durch das Setzen der so genannten UVW-Koordinaten (Bildkoordinaten) definiert. Zumeist ist es möglich die Originalgröße der Texturbilder anzugeben (z. B. Modifikator UVW Map im 3D Studio Max) und diese Werte sind bei jeder Fläche, die mit dem Material belegt sind, jeweils gleich zu setzen.

4.2.2 Hinzufügen externer Projektdaten


Durch Klicken auf den Button `Externe Projektdaten importieren`  in der Haupt-Toolleiste können externe Daten importiert werden. Folgende Formate werden unterstützt:

- AutoCAD *.dwg oder *.dxf
- 3D Studio *.max oder *.3ds
- VRML Virtual Reality Modeling Language *.wrl
- CityGRID® Planner Projekte *.cgp

Nach dem Auswählen einer Datei erscheint der 3D-Studio-Dialog zum Import von Daten, der je nach Datenformat unterschiedlich ist.



Hinweis: Falls vorhanden, die **Checkbox „Reset Scene“ unbedingt deselektieren**, damit die bisherigen Projektdaten nicht verloren gehen!


Durch Klicken auf  wird das Modell mit seinem Namen in die aktive Szene aufgenommen.




Hinweis: Achten Sie darauf, dass die externen Daten im selben reduzierten Koordinatensystem (Projektkoordinatensystem) vorliegen, wie das Builder-Projekt (vgl. 3.1.1). Die Daten sollten vorweg in das Projektkoordinatensystem transformiert werden, da nachträgliche Verschiebungen im 3D Studio zu Koordinatenverfälschungen führen können.

4.2.3 Hinzufügen von Projektdaten im 3D Studio

Objekte, die im 3D Studio MAX direkt erstellt werden, werden erst beim Update des Projektexplorers (durch

Klicken auf den Button `Ansicht aktualisieren`  der Haupt-Toolleiste) in das Builder-Projekt unter der aktiven Szene aufgenommen. Dabei wird der Name des Objekts von 3D-Studio übernommen und im Projektexplorer eingetragen.



Tipp: Externe oder intern erstellte Projektdaten können einfach mittels des Buttons `Auf das Gelände setzen`  der Haupt-Toolleiste (vgl. III 6.1) höhenmäßig an ein vorhandenes CityGRID® Geländemodell angepasst werden. Wenn kein CityGRID® Geländemodell verfügbar ist, bleibt die Höhe unverändert.

4.3 Bibliotheksdaten

Der CityGRID® 3D Studio Builder bietet die Verwendung von Bibliotheken für 3D Objekte an. In einer Bibliothek werden typischerweise wiederkehrende 3D Objekte eines oder mehrerer Builder-Projekte abgelegt. (z.B. Straßenmobiliar, Vegetation, Autos etc.)

4.3.1 Aufbau einer Bibliothek

Die Definition der Bibliotheken obliegt dem Anwender selbst. In den 3D Studio Builder wird nur der Verweis auf den Speicherort der 3D Modell selbst aufgenommen. Einmal angelegte Bibliotheken stehen fortan in jedem 3D Studio Builder zur Verfügung.

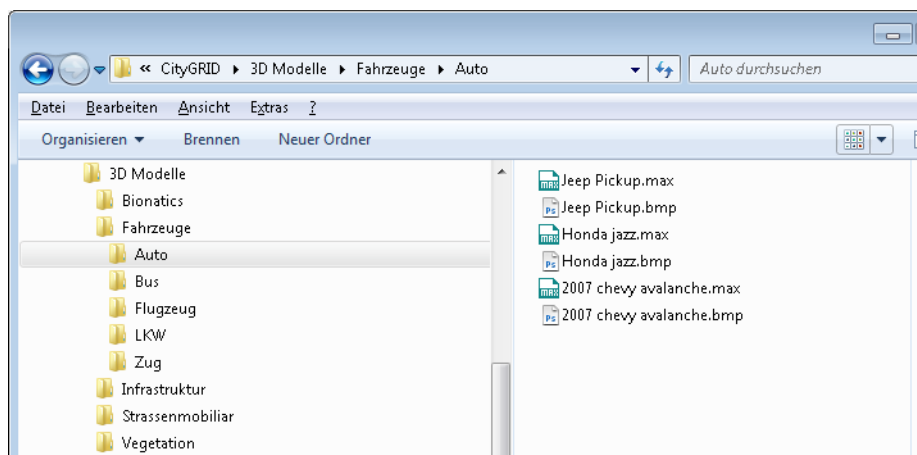
Bibliotheken müssen folgenden Aufbau haben:

- Je Bibliothek ist ein eigenes Verzeichnis anzulegen. Der Verzeichnisname definiert den Namen der Bibliothek. (z.B. Fahrzeuge)
- Darunter ist zumindest ein weiteres Verzeichnis zu erstellen. Jedes Verzeichnis begründet eine Kategorie der Bibliothek. (z.B. Auto, LKW, Eisenbahn, Schiff)
- In diesem Verzeichnis werden nun die eigentlichen 3D Daten im Format .max abgelegt. Jede .max Datei stellt einen Bibliotheksobjektyp dar.



Hinweis: Die jede .max Datei muss folgende Kriterien erfüllen um als Bibliotheksobjekt Verwendung finden zu können:

- Die Geometrie liegt als Vielflächennetz (Mesh) vor.
 - Materialien sind als Typ Standard definiert und verwenden Texturbilder nur in der Streufarbe (Diffuse Color).
 - Der Ursprung des Koordinatensystems liegt in X/Y mittig zum Objekt, in Z am tiefsten auftretenden Vertex.
 - Die Maßeinheit ist identisch mit jenen der anderen 3D Daten der Szene (typischerweise Meter).
- Zu jeder .max Datei kann es eine gleichnamige .bmp Datei geben, die ein Vorschaubild des 3D Objekts zeigt. Die .bmp Datei muss mit einer Breite von 95 Pixel und einer Höhe 85 Pixel erstellt werden.




Wenn die oben beschriebene Ordnerstruktur angelegt wurde, kann zu jeder Zeit eine Erweiterung, Änderung oder Reduktion der Bibliotheksobjektypen erfolgen. Jede Änderung an den .max Dateien wirkt sich umgehend auf den 3D Studio Builder aus.

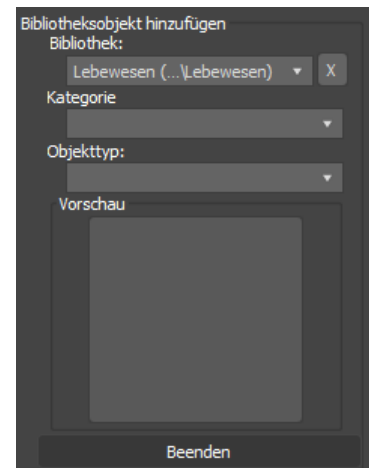


Hinweis: Änderungen an den .max Dateien haben keine Auswirkungen auf bereits gesetzte Bibliotheksobjekte in Szenen.

4.3.2 Neue Bibliothek anlegen

Um eine neue Bibliothek einzubinden, sind folgende Schritte notwendig:

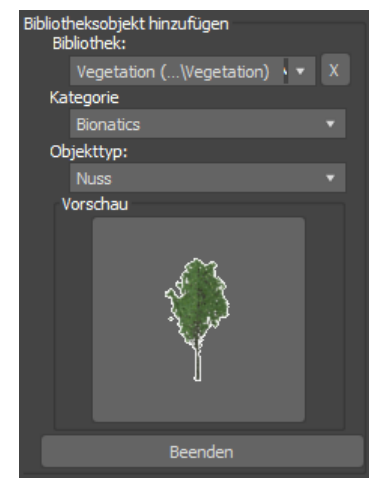
1. Bibliotheksmodus starten: In der Haupt-Toolleiste den Button  **Bibliothekssobjekte hinzufügen** klicken. Es wird das Fenster *Bibliothekssobjekt hinzufügen* unter dem Projektextplorer angedockt.
2. In der Dropdownliste von *Bibliothek* den Eintrag *Bibliothekssordner hinzufügen* wählen. Sobald dieser angeklickt wurde öffnet sich ein Dateibrowser in dem zum Ordner der einzubindenden Bibliothek (vgl. III 4.3.1) navigiert wird.
3. Durch Klicken auf **OK** wird der Ordner eingebunden und die Daten in diesem Ordner stehen als neue Bibliotheksobjekte zur Verfügung.

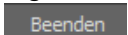


4.3.3 Bibliotheksobjekte in Szene einfügen

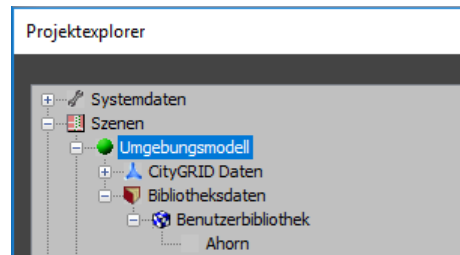
Das Hinzufügen eines Geometrieobjekts aus einer Bibliothek erfolgt folgendermaßen:

1. Bibliotheksmodus starten: In der Haupt-Toolleiste den Button  **Bibliothekssobjekte hinzufügen** klicken. Es wird das Fenster *Bibliothekssobjekt hinzufügen* unter dem Projektextplorer angedockt.
2. *Bibliothek* auswählen: Zur Auswahl stehen alle zuvor eingebundenen Bibliotheksordner (vgl. III 4.3.2).
3. *Kategorie* auswählen: Falls eine Bibliothek mehrerer Kategorien besitzt erscheinen diese in der Auswahlliste. Wenn nur eine Kategorie angelegt wurde, wird diese automatisch gewählt.
4. *Objekttyp* aus der Bibliothek wählen: Aus der Auswahlliste das gewünschte Objekt wählen. Das Vorschau-Fenster zeigt ein Muster des gewählten Typs falls diesem ein Vorschaubild zugewiesen wurde (vgl. III 4.3.1).
5. Durch Klicken auf die *Vorschau* wird der Einfügemodus gestartet. Solange der Einfügemodus aktiv ist, ist die Schaltfläche gelb hinterlegt. Die jeweils vom Benutzer erwartete Eingabe im Viewport wird in der Statuszeile angegeben.
6. Neues Objekt hinzufügen und positionieren: Der erste Klick mit der linken Maustaste in einem der Grafikenster ist immer die 2D-Positionierung des Referenzpunkts des neu hinzugefügten Bibliotheksobjekts. Wurde das Objekt innerhalb eines allfällig vorhandenen CityGRID® Geländemodells positioniert, wird das Objekt automatisch auf das Gelände gesetzt. Das Hinzufügen neuer Objekte kann mit einem Rechtsklick im Grafikenster abgebrochen werden.
7. Parameterauswahl: Durch weitere Klicks können die Parameter Drehung, Länge, Breite und Höhe eines Objekts gesetzt werden. Der momentan zu setzende Parameter wird in der Statusleiste angezeigt. Dabei gelten folgende Regeln:
 - Befindet sich der Mauszeiger in der Nähe des letzten Mausklicks, wird der Standardwert für den Parameter verwendet. Diese Position wird durch ein Linienkreuz angezeigt.
 - Je weiter man sich mit dem Mauszeiger von der Position des letzten Mausklicks entfernt, desto größer/kleiner wird der Wert des Parameters. Der jeweils verwendete Parameterwert wird in der Statuszeile angezeigt.
 - Drückt man die linke Maustaste, wird dieser Wert übernommen und die Parameterauswahl springt zum nächsten Parameter.
 - Drückt man die rechte Maustaste, wird dieser Wert übernommen und die Parameterauswahl wird abgebrochen (die verbleibenden Parameter werden mit Standardwerten gesetzt).



8. Nächstes Objekt: Nachdem alle Parameter gesetzt wurden bzw. die Parameterauswahl abgebrochen wurde, kann mit dem nächsten Mausklick bereits das nächste Objekt gesetzt werden. Weiter bei Punkt 5 bzw. mit einem Rechtsklick das Einfügen eines neuen Objekts abbrechen.
9. Das Einfügen von Bibliotheksdaten durch Klicken auf den Button  abschließen.

Das Fenster *Bibliotheksojekt hinzufügen* wird geschlossen und der Projektextplorer wird aktualisiert. Die eingefügten Bibliotheksdaten erscheinen im Projektextplorer in der aktiven Szene unter dem Knoten *Bibliotheksdaten*. Durch Auswahl des Objekts im Projektextplorer wird sein Eigenschaftsfenster angedockt, sodass Parameter auch noch im Nachhinein numerisch angepasst werden können.



4.3.4 Bibliotheksdaten mittels externer Steuerdatei hinzufügen

Um eine Fülle von Bibliotheksdaten in einem Schritt hinzufügen zu können, kann eine Steuerdatei vorbereitet werden, in der die wesentlichen Informationen für jedes Objekt angegeben sind, wie die Bibliothekskategorie, Position und Skalierung.

Die Steuerdatei ist eine ASCII-Datei, in der für jedes einzufügende Objekt eine Zeile angegeben wird. In jeder Zeile stehen die Parameter durch Leerzeichen oder Tabulatoren getrennt.

Folgende Parameter können angegeben werden:

- X, Y (erforderlich): die Position des Bibliotheksobjekts, Als Dezimaltrennzeichen ist der Punkt (.) zu verwenden!
- Z (optional): die Position in Höhe des Bibliotheksobjekts kann automatisch ermittelt werden.
- L, B, H (Länge, Breite, Höhe) oder H, D (Höhe, Durchmesser): Größenwerte in Meter für die Ausdehnung des Bibliotheksobjekts.
- BN: Bibliotheksname
- BK: Bibliothekskategorie
- K: Kategorie der Bibliothek



***Hinweis** Falls ein Objekt eines bestimmten Objekttyps in der angegebenen Kategorie nicht gefunden wird, erfolgt keine Platzierung der Objekte. Der Bindestrich (-) kann als Wildcard verwendet werden um alle Bibliotheken und Kategorien zu durchsuchen. Bei Namensgleichheit von Bibliotheken oder Kategorie wird der erste gefundene Eintrag verwendet*

- BO: Name des Objekttyps

Wir empfehlen, diese Steuerdatei im Referenzkoordinatensystem zu führen, damit sie für verschiedene Projekte verwendet werden kann (vgl. III 3.1.1)



***Beispiel:** Eine Steuerdatei der Formatierung [X][Y][L][B][H][BN][BK][BO]*


567898	3678909	5	8	12	Straßenraum	Beleuchtung	Laterne
567887	3678910	5	8	12	Straßenraum	Beleuchtung	Laterne
567895	3678912	5	8	12	Straßenraum	Beleuchtung	Laterne

Es werden drei Bibliotheksobjekte „Laternen“ an den Punkten 567898/3678909; 567887/3678910; 567895/3678912 platziert. Die Laternen haben eine Länge von 5m, eine Breite von 8m und eine Höhe von 12m. Zu finden ist das Bibliotheksobjekt „Laterne“ in der Kategorie „Beleuchtung“ der Bibliothek „Straßenraum“

Beim Hinzufügen von Bibliotheksobjekten kann:


- die Höhenposition der Objekte automatisch ermittelt werden.
- die Fülle der Bibliotheksobjekte der Steuerdatei über eine Gebietsdefinition eingeschränkt werden. Dadurch kann eine einzige derartige Steuerdatei für verschiedene Szenen wiederverwendet werden, wobei nur die Objekte, die in der jeweiligen Szenenausdehnung liegen, zum Projekt hinzugefügt werden.
- einen anderen Koordinatenoffset als den Projektoffset berücksichtigen:

Das automatisierte Einfügen von Bibliotheksobjekten erfolgt folgendermaßen:

1. In der Haupt-Toolleiste den Button Bibliotheksobjekte  mittels externer Datendatei positionieren klicken. Es werden zwei Fenster unter dem Szene-Explorer angedockt:

- eines zur Angabe der Datenquelle und
- eines zur Angabe von Einstellungen.

2. Durch Klicken auf die erste Schaltfläche öffnet sich ein Dateibrowser, in dem eine Steuerdatei ausgewählt werden kann. Die Formatierung dieser Datei muss durch Auswahl der entsprechenden *Formatierungsvorlage* angegeben werden. Durch Klicken auf den

Info-Button  wird eine Kurzbeschreibung der Abkürzungen gezeigt.

3. *Automatische Höhenpositionierung:*


Falls aktiviert, wird die Z-Position der eingefügten Bibliotheksobjekte automatisch ermittelt (und ev. vorhandene Z-Koordinaten der Steuerdatei ignoriert). Folgende Ermittlungsvorschriften stehen zur Auswahl:

- CityGRID Gelände: Das Objekt wird auf in der Szene vorhandene CityGRID®-Geländemodelle gesetzt.
- aktuelle Auswahl: Alle selektierten Flächenobjekte werden verwendet, wobei die höchste ermittelte Z-Koordinate bei überlappenden Flächen verwendet wird.
- alle Flächenobjekte: Alle in der Szene vorhandenen Flächenobjekte werden verwendet, wobei die höchste ermittelte Z-Koordinate bei überlappenden Flächen verwendet wird.



***Tipp:** Im Builder Control Center lässt sich die Höhenpositionierung auch über die Config Seite umsetzen. (vgl. Handbuch Builder Control Center)*

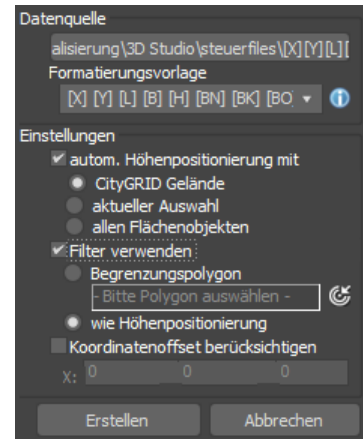
4. *Filter verwenden:*

Hier kann eine Gebietseingrenzung vorgenommen werden. Nur jene Objekte, die innerhalb des angegebenen Polygons (das mittels des Pick-Buttons  selektiert werden kann) bzw. für die eine gültige Höhe berechnet werden konnte, werden eingefügt.

5. *Koordinatenoffset berücksichtigen:*

Falls die Steuerdatei Koordinaten in einem anderen System als dem Referenzkoordinatensystem enthält, kann der Offset (zum Referenzkoordinatensystem (!) und nicht zum Projektkoordinatensystem) angegeben werden.

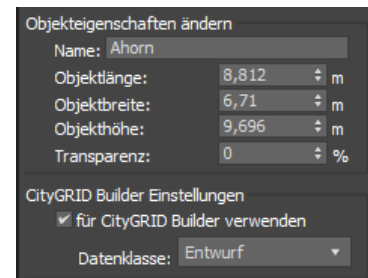
6. Durch Klicken auf  werden die Bibliotheksobjekte zum Projekt hinzugefügt.




4.3.5 Ändern von Bibliotheksdaten

Selektiert man ein Bibliotheksobjekt im Projektexplorer, werden seine Parameter im Eigenschaftsfenster angezeigt. Die Eigenschaften des Objekts können hier geändert werden und werden sofort (oder nach Bestätigen mit Return) übernommen.

Geändert werden können neben dem Namen des Objekts die Grundeinstellungen Objektlänge, -breite, -höhe und Transparenz, die für alle Bibliotheksobjekte gelten.




4.4 Geometriedaten aus dem Projekt löschen

Daten können durch Drag & Drop im Projektexplorer auf den Papierkorb  (vgl. III 2.1) aus dem Projekt entfernt werden.



Hinweis: Um den Inhalt des Papierkorbs endgültig zu löschen steht im Menü „CityGRID Builder“ der 3D Studio Max Hauptmenüleiste die Funktion Papierkorb löschen zur Verfügung. Durch Ausführen dieses Menüpunktes leert sich der Inhalt des Papierkorbs im Projektexplorer und die zugehörigen Backupdateien im Builder-Projekt werden gelöscht.

4.4.1 Gelöschte Daten wiederherstellen

Solange der Papierkorb  nicht gelöscht wurde, können Daten aus diesem wieder in das Projekt verschoben werden. Dazu ist der wiederherzustellende Eintrag mittels drag & drop aus dem Papierkorb auf den jeweiligen Hauptknoten (Systemdaten oder Szenen) zu verschieben.



Hinweis: Zu beachten ist, dass Szenen zwar gelöscht, aber nicht gesamt wiederhergestellt werden können. Beim Löschen einer Szene werden alle enthaltenen Daten in den Papierkorb verschoben, die Szene Datei selbst aber vernichtet. Um die Daten einer Szene wiederherzustellen ist daher eine neue Szene anzulegen, oder eine bereits bestehende Szene auszuwählen, die die Daten aufnehmen soll.

5 Systemdatenhinzufügen / ändern / entfernen



Systemdaten wie Kamera und Sonnenstand werden global mit dem aktuellen Builder-Projekt mit gespeichert und sind von den momentan aktiv geladenen Szenen unabhängig.

5.1 Kamera

Kameras definieren Blickpunkte in die 3D Szene. Sie können als fixe Ausgangspunkte für die Erkundung des Scout-Projekts dienen, aber auch als animierte Rundgänge eingesetzt werden.

Jede Kamera wird durch zwei Punkte bzw. dem dazwischen aufgespannten Vektor definiert: Dem Kamerapunkt und dem Zielpunkt. Der Kamerapunkt entspricht der Lage im 3D Raum, auf dem sich die Kamera tatsächlich befindet. Der Zielpunkt hingegen ist jener Punkt, auf dem der Fokus der Kamera liegt. Der Richtungsvektor zwischen Kamera- und Zielpunkt legt dadurch die Blickrichtung und den aufgenommenen Teil der 3D Szene fest.



Der 3D Studio Builder bietet 4 Typen von Kameras bzw. Ansichten :

- **Fixe Kamera** : Bei einer fixen Kamera sind sowohl der Kamera- als auch der Zielpunkt fixiert. Dieser Typ wird zur Definition von fixen Blickpunkten im Scout eingesetzt.
- **Relative Kamera** : Bei einer relativen Kamera ist die Blickrichtung fixiert. Ausgehend vom aktuellen Zielpunkt des Ansichtsfensters wird der Kamerapunkt entlang der gespeicherten Blickrichtung zurückversetzt.

Wechselt man zwischen verschiedenen relativen Kameras, bewegt sich der Kamerapunkt auf einer Kugel mit festem Radius um den aktuellen Zielpunkt des Ansichtsfensters.

Wechselt man im Projektexplorer zu einem Geometriedaten-Blatt (ein Blatt ist ein Knoten im Baum, der keinen Unterknoten besitzt), wird diese virtuelle Kugel in den Schwerpunkt des neuen aktiven Knotens verschoben und zusätzlich ein Zoom auf dessen Ausdehnungen gemacht.

Relative Kameras können nicht in ein CityGRID® Scout Projekt exportiert werden. Sie dienen nur der Szenenaufbereitung im 3D Studio.

- **Rundflug** : Bei einem Rundflug wird der Kamerapunkt entlang eines Pfades fortbewegt, der Zielpunkt aber konstant gehalten wird. Dadurch dreht sich die Kamera um einen fix eingestellten Punkt.
- **Animation** : Bei einer Animation werden sowohl der Kamerapunkt als auch der Zielpunkt entlang eines Pfades fortbewegt. Beide Punkte können einen eigenständigen Pfad zugewiesen bekommen. Dadurch lassen sich Flüge durch die 3D Szene umsetzen.


Beim Anlegen eines neuen Builder-Projekts werden 5 vordefinierte relative Kameras angelegt: die Ansicht aus den vier Himmelsrichtungen sowie die Draufsicht.

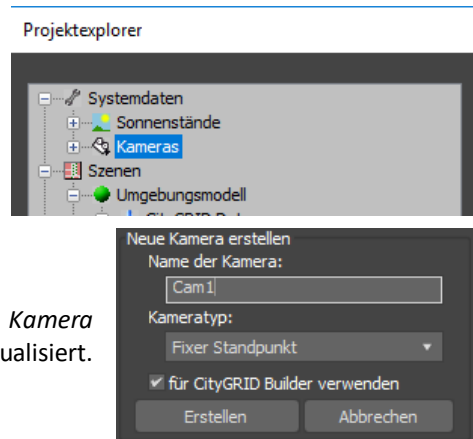


Hinweis: Kameraanimationen und Blickpunkte lassen sich auch direkt im CityGRID® Scout erstellen (vgl. Handbuch CityGRID® Scout)

5.1.1 Neue Kamera erstellen

Neue Kameras werden durch Selektion des Knotens *Kamera* (unter *Systemdaten*) im Projektexplorer angelegt. Sobald *Kamera* angeklickt wurde öffnet sich ein Eigenschaftsfenster, in dem die wichtigsten Merkmale der neuen Kamera gesetzt werden können.

1. Geben Sie der neuen Kamera einen Namen
2. Wählen Sie einen *Kameratyp* (vgl. III 5.1)
3. Klicken Sie auf . Das Fenster *Neue Kamera erstellen* wird geschlossen und der Projektexplorer wird aktualisiert.



Die neu erstellte Kamera erscheint im Projektexplorer unter **Systemdaten** – **Kameras** und wird automatisch ausgewählt. Dadurch wird ihr Eigenschaftsfenster angedockt, sodass weitere Parameter eingestellt werden können (vgl. III 5.1.2).

Je nach Kamera-Typ wird die neue Kamera mit folgenden Einstellungen erstellt:

- Für fixe Kameras werden der aktuell im 3D Perspektivfenster eingestellte Kamera- und Zielpunkt übernommen.
- Für relative Kameras wird die aktuell im 3D Ansichtsfenster eingestellte Blickrichtung übernommen.
- Für einen Rundflug wird ein kreisförmiger Kamerapfad rund um den aktuell im 3D Perspektivfenster eingestellten Zielpunkt (Fokus der Perspektive) durch den aktuellen Kamerapunkt gelegt.
- Für eine Animation werden ein Kamerapunkts- und ein Zielpunkts- Pfad, die jeweils durch einen $\frac{3}{4}$ -Kreis repräsentiert sind, angelegt und Kamera bzw. Ziel daran gebunden. Der Zielpunkts- Pfad ist identisch zu jenem des Kamerapunktes.



Hinweis: Die Pfade für Rundflug und Animation sind als Vorschläge zu betrachten, und können vom Anwender jederzeit verändert, oder neu angelegt werden. Über 3D Studio Konstruktionsmethoden sind dazu Linienzüge anzulegen und den jeweiligen Kameras zuzuweisen.

5.1.2 Kameraeinstellungen ändern

Wird eine Kamera im Projektexplorer oder über die Schnellauswahl im Hauptfenster selektiert, wird die 3D-Ansicht mit den Kameraeinstellungen aktualisiert.

Wird eine Kamera im Projektexplorer selektiert, öffnet sich auch das Eigenschaftsfenster der Kamera. Hier können Einstellungen geändert bzw. nach dem Neuerstellen weitere Einstellungen vorgenommen werden:


- Für fixe und relative Kameras kann die Ansicht im 3D Perspektivfenster angepasst und durch Klicken auf **Änderungen übernehmen** neu gespeichert werden.

- Für Rundflüge und Animationen:

- **Brennweite der Kamera:**
50 mm entspricht Normalwinkel, 35 und 28 sind Weitwinkel- bzw. Superweitwinkelbrennweiten, Werte größer 50 sind Telebrennweiten.

- **Horizontales Blickfeld:**
der Öffnungswinkel, der sich aus oben eingestellter Brennweite ergibt.

- **Länge der Animation:**
die Dauer des Rundflugs bzw. der Animation

- **Pfad der Kamera:**
Der Kamerapfad wird im 3D Perspektiv- als auch im Grundrissfenster angezeigt und kann dort geändert, oder neu erstellt werden. Mittels des Pick-Buttons  rechts des Pfadnamefeldes kann auch eine beliebige Linie als neuer Kamerapfad ausgewählt werden (die Aufforderung, den Pfad zu selektieren erscheint in der Statuszeile).





Hinweis: Die Vorgangsweise zur Erstellung neuer Pfade entnehmen kann der 3D Studio Max Hilfe entnommen werden. Neu erstellte Pfade dürfen nicht in das Builderprojekt übernommen werden, die Option „für CityGRID® Builder verwenden“ des Geometrieobjekts im Projektexplorer unter Projektdaten muss entfernt werden.

- **Pfad des Zielpunkts:** Für Animations-Kameras kann zusätzlich zum Kamerapfad auch ein Pfad des Zielpunkts angegeben werden.

Sämtliche Änderungen werden erst durch Klicken auf **Änderungen übernehmen** wirksam.

5.2 Sonnenstände

CityGRID® 3D Studio Builder unterstützt sowohl statische  als auch dynamische Sonnengänge . Damit können einerseits Schattenbilder gerendert und andererseits Schattenanimationen im CityGRID® Scout generiert werden.

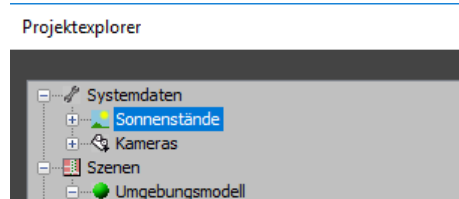


Hinweis: Schattenwürfe sind im CityGRID® Scouts ohne Vorberechnungen möglich.

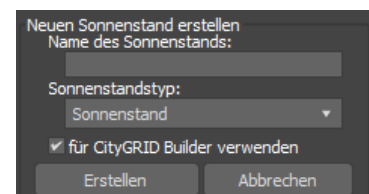
Beim Anlegen eines neuen Builder-Projekts werden 4 vordefinierte Sonnenstände und 4 Sonnengänge zum jeweiligen Jahreszeitenwechsel vorbereitet.


5.2.1 Neuen Sonnenstand erstellen

Neue Sonnenstände bzw. -animationen werden durch Selektion des Knotens *Sonnenstände* (unter *Systemdaten*) im Projektextplorer angelegt. Sobald *Sonnenstände* angeklickt wurde öffnet sich ein Eigenschaftsfenster, in dem die wichtigsten Merkmale des neuen Sonnenstandes gesetzt werden können.



1. Geben Sie dem neuen Sonnenstand/-gang einen Namen
2. Wählen Sie den *Sonnenstandstyp* (vgl. III 5.2)
3. Klicken Sie auf *Erstellen*. Das Fenster *Neuen Sonnenstand erstellen* wird geschlossen und der Projektextplorer wird aktualisiert.

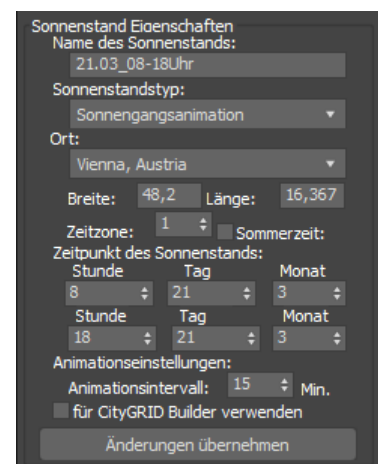


Der neu erstellte Sonnenstand erscheint im Projektextplorer unter *Systemdaten* – *Sonnenstände*  und wird automatisch ausgewählt. Dadurch wird sein Eigenschaftsfenster angedockt, sodass weitere Parameter eingestellt werden können (vgl. III 5.2.2).

5.2.2 Sonnenstand Einstellungen ändern


Wird ein Sonnenstand im Projektextplorer selektiert, öffnet sich auch das Eigenschaftsfenster des Sonnenstands. Hier können Einstellungen geändert bzw. nach dem Neuerstellen weitere Einstellungen vorgenommen werden:

- *Ort:*
Standardmäßig wird der bei der Installation der Software angegebene Ort verwendet. Die geografische *Breite* und *Länge* sowie die *Zeitzone* des Ortes werden automatisch angepasst, wenn ein anderer Ort aus der Liste ausgewählt wird. Daneben können diese Daten aber auch interaktiv angepasst werden.
- *Sommerzeit:*
Geben Sie an, ob an diesem Ort zum Zeitpunkt des Sonnenstands Sommerzeit gilt.
- *Zeitpunkt des Sonnenstands:*
Geben Sie Datum und Uhrzeit an.
- Für Sonnengangsanimationen: Geben Sie in der zweiten Zeit-Zeile den Endpunkt der Sonnengangsanimation an. Weiters muss das *Animationsintervall* eingegeben werden, das Zeitintervall in dem für eine Animation der neue Schatten berechnet werden soll.



Sämtliche Änderungen werden erst durch Klicken auf  wirksam.


5.3 Systemdaten von Projekt entfernen

Daten können durch Drag & Drop im Projektextplorer auf den Papierkorb  (vgl. III 3.1) aus dem Projekt entfernt werden. Davon ausgenommen sind die fünf relativen Kameras *NordAnsicht*, *OstAnsicht*, *SüdAnsicht*, *WestAnsicht* und *Draufsicht*, sowie der Sonnenstand *Ausgeschaltet*.

6 Tools


6.1 Objekte auf das Gelände setzen

Geometriedaten können auf einfache Weise in der Z-Richtung auf das Geländemodell verschoben werden. Voraussetzung: Es existiert ein Geländemodell in den CityGRID® Daten, unterhalb des Einsetzpunktes der Objekte.

1. Selektieren Sie im Grafikfenster das/die zu verschiebende(n) Objekt(e).
2. Klicken Sie auf den Button **Objekte auf Gelände setzen**  der Haupt-Toolleiste. Es wird für jedes des selektierten Objekte die Lotrechte durch den Einsetzpunkt (Zentrum des Objektkoordinatensystems) gelegt und entlang dieser Lotrechten der tiefste Lotpunkt ermittelt. Dieser tiefste Lotpunkt liegt für Bibliotheksobjekte aus Systembibliotheken im Allgemeinen in der Mitte des Objekts, kann bei anderen Objekten aber auch am Rand liegen. Das Objekt wird schließlich so verschoben, dass dieser tiefste Lotpunkt auf dem CityGRID® Geländemodell zu liegen kommt.

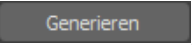
6.2 Objekte vervielfältigen

Geometriedaten können auf einfache Weise vervielfältigt werden. Wird beispielsweise ein Baum entlang eines Pfades vervielfältigt, kann auf einfache Weise eine Allee generiert werden.

1. Selektieren Sie das zu vervielfältigende Objekt im Viewport.
2. Klicken Sie auf den Button **Objekte vervielfältigen**  in der Haupt-Toolleiste. (vgl. III 2.1) Ein Einstellungsfenster öffnet sich. Anhand der aktuellen Einstellungen werden die Positionen der neu hinzuzufügenden Objekte durch blaue Kreuze angezeigt.
3. *Methode:*
Wählen Sie als Methode Orthogonalversatz, Ringversatz oder Pfadversatz, nach der das selektierte Objekt vervielfältigt werden soll (vgl. nachfolgende Abschnitte)
4. *Typ:*
Wählen Sie den Typ, wie die neu erzeugten Objekte eingefügt werden sollen. Dieser Typ stellt die Möglichkeiten zur Verfügung, wie Objekte in 3D Studio Max verwaltet werden können.
 - *Eigenständige Kopien:*
Die neu erzeugten Objekte werden als unabhängige Kopien des Originals generiert.
 - *Instanzobjekte:*
Eine Instanz ist eine veränderbare Kopie des Originals. Ändern der Instanz ist gleichbedeutend mit Ändern des Originals (empfehlenswert bei hoher Anzahl gleich aussehender Objekte)
 - *Referenzobjekte:*
Referenzen sind „Einbahn-Instanzen“. Änderungen am Original werden auch für die Referenz übernommen. Hingegen werden Änderungen an der Referenz nicht auf das Original übertragen (empfehlenswert, wenn jedes Objekt individuell angepasst wird, trotzdem aber alle auf einmal grundlegende Änderungen erfahren sollen, etwa einen anderen Baumtyp).



Hinweis: Für eine genaue Beschreibung von Instanzen und Referenzen verweisen wir auf die Hilfe von 3D Studio.

5. Klicken Sie auf **Generieren** , um an den in der Vorschau angezeigten Positionen neue Objekte zu erzeugen.

6.2.1 Orthogonalversatz

Die neu erzeugten Objekte werden in einem regelmäßigen Raster generiert:

- **Anzahl Dim 1:**
die Anzahl der neu zu erzeugenden Objekte in ersten Dimension.
- **x/y-Abstand:**
die Richtung der ersten Dimension.
- **Drehung:**
der Winkel, um den das jeweilige Objekt zu seinem Vorgänger im Gegenuhrzeigersinn verdreht wird. Wird beispielsweise ein Winkel von 10° eingegeben, wird das erste neu erzeugte Objekt um 10° verdreht, das zweite bereits um 20° usw.
- **Anzahl Dim 2: x/y-Abstand und Drehung:**
Auf gleiche Weise kann in eine zweite Dimension vervielfältigt werden.
- Drehungen aus der 1. und 2. Dimension werden addiert.



Hinweis: Im Beispiel der Abbildung werden 11 neue Objekte generiert, denn in der 1. Dimension werden 3 neue erzeugt (ergibt 4 Objekte in dieser Dimension) und in der 2. Dimension werden 2 neue erzeugt (ergibt 3 Objekte in dieser Dimension). In Summe ergeben sich damit 12 Objekte (11 neue).

6.2.2 Ringversatz

Die neu erzeugten Objekte werden in einem Kreis durch das Originalobjekt generiert.

- **Anzahl:**
die Anzahl der neu zu erzeugenden Objekte auf dem Kreis. Sie werden gleichmäßig im Kreis aufgeteilt.
- **Radius:**
Radius des Kreises
- **Startwinkel:**
Startwinkel der Tangente des Kreises durch die Position des Originalobjekts.



Hinweis: Die Winkelwerte bedeuten:

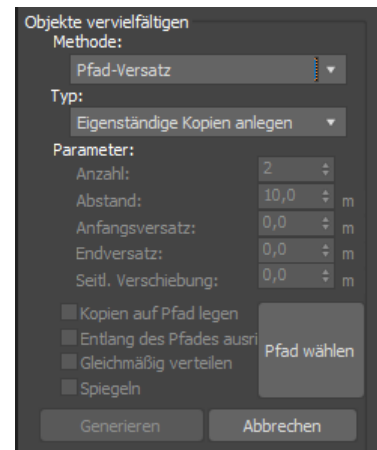
0: Tangente parallel zur Y-Richtung, der Kreis wird links des Originals gelegt.
90: Tangente parallel zur X-Richtung, der Kreis wird unterhalb des Originals gelegt.
180: Tangente parallel zur Y-Richtung, der Kreis wird rechts des Originals gelegt.
270: Tangente parallel zur X-Richtung, der Kreis wird oberhalb des Originals gelegt.
Zusätzlich ist jeder Zwischenwinkel möglich.

- **Anzahl Ringe, Ringabstand:**
Vom ersten Kreis ausgehend können weitere konzentrische Kreise generiert werden, wobei der Radius jeweils um den Ringabstand vergrößert wird. Die Anzahl der erzeugten Objekte bezieht sich bei mehreren Kreisen auf die Anzahl pro Kreis.
- **Zum Mittelpunkt ausrichten:**
Damit kann gefordert werden, dass jedes Objekt so verdreht wird, dass alle Objekte dieselbe Seite zum Mittelpunkt ausrichten.

6.2.3 Pfadversatz

Die neu erzeugten Objekte werden entlang eines Pfades generiert. Klicken Sie zunächst auf den Button **Pfad wählen** und selektieren Sie danach einen Pfad in einem der Grafikfenster.

- **Anzahl, Abstand:**
die Anzahl und der Abstand der neu zu erzeugenden Objekte entlang des Pfads.
- **Anfangsversatz:**
der Abstand des ersten zu erzeugenden Objekts vom Startpunkt des Pfads (gemessen entlang des Pfads).
- **Endversatz:**
der Abstand des letzten zu erzeugenden Objekts vom Endpunkt des Pfads (gemessen entlang des Pfads); nur auswählbar, wenn „Gleichmäßig“ aktiv ist.
- **Seitl. Verschiebung:**
positiver oder negativer Normalabstand der zu erzeugenden Objekte vom selektierten Pfad; nur auswählbar, wenn „Kopien auf Pfad legen“ aktiv ist.
- **Kopien auf Pfad legen:**
gibt an, ob die neu zu erzeugenden Objekte auf dem Pfad selbst oder auf einer Parallelen des Pfads durch die Position des Originalobjekts generiert werden.
- **Entlang des Pfades ausrichten:**
gibt an, ob die neu zu erzeugenden Objekte mit der Tangentenrichtung des Pfads mitgedreht werden sollen.
- **Gleichmäßig verteilen:**
gibt an, ob die neu zu erzeugenden Objekte gleichmäßig entlang des Pfads aufteilt werden sollen. In diesem Fall kann der Anfangs- und Endversatz angegeben werden, nicht jedoch der Abstand (weil dieser automatisch ermittelt wird).
- **Spiegeln:**
Falls eine seitliche Verschiebung angegeben ist, können die Objekte verdoppelt werden, indem sie zusätzlich um den Pfad gespiegelt generiert werden.




Beispiel: Die Generierung einer doppelten Baumreihe einer Allee mit der Straßenachse als Pfad ist derart möglich.

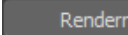
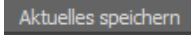

6.3 Ansicht aktualisieren

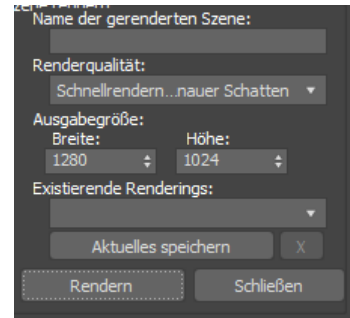


Durch Klicken auf den Button **Ansicht aktualisieren** in der Haupt-Toolleiste werden einerseits alle Ansichtsfenster neu gezeichnet und andererseits der Projektextplorer aktualisiert. Dies ist notwendig, wenn im Viewport neue Objekte gezeichnet wurden.

7 Animationen und Bilder rendern

Durch Klicken auf den Button **Animationen und Bilder rendern**  der Haupt-Toolleiste wird das Optionsfenster an den Projektextplorer angedockt. Für das Rendern werden die aktuelle Kameraeinstellung und die in der Schnellauswahl aktivierte Sonnenstands-Einstellung verwendet.

1. Selektieren Sie das Ansichtsfenster, von dem Sie das Video / Bild rendern wollen
2. Geben Sie einen *Namen* für die gerenderte Animation/Ansicht ein.
3. Wählen Sie eine *Renderqualität* für die Schatten (je höher die Qualität, desto länger die Renderzeit)
4. Definieren Sie die *Ausdehnungen* der gerenderten Animation/Ansicht.
5. Durch Klicken auf **Rendern**  öffnet sich das Renderfenster von Autodesk 3D Studio MAX, in dem das Bild zu sehen ist bzw. in dem die gerenderten Frames zu sehen sind. Sie können den Renderprozess durch Drücken der „Esc“-Taste abbrechen (das Ergebnis, das bis dahin gerendert wurde bleibt erhalten).
6. Wird ein Video gerendert, erscheint am Ende der Video Player zur Betrachtung des Videos.
7. Wiederholen Sie Schritte 3 bis 6 solange, bis das Ergebnis Ihren Anforderungen entspricht.
8. Drücken Sie auf **Aktuelles speichern**  um das gerenderte Ergebnis zu speichern. Die Animation / Ansicht wird dann im Projektverzeichnis im Unterordner „Media“ angelegt. Weiters wird diese Animation/Ansicht in die Liste der existierenden Renderings aufgenommen, sodass sie später schnell wieder geladen werden kann. Durch Klicken auf „X“  wird das aktuelle Rendering gelöscht.



Hinweis: Falls sowohl eine Rundflugs- oder Animationskamera als auch eine Sonnengangsanimation aktiv sind, wird die Dauer der gerenderten Animation nach der Sonnengangsanimation bestimmt. Um zu gewährleisten, dass in dieser Zeit die Kamera den gesamten Animationspfad durchläuft, muss bei den Einstellungen der Kamera die Animationsdauer auf dieselbe Zeit gesetzt werden, wie sie sich aus der Sonnengangsanimation (Zeitspanne und Intervall) ergibt.



Hinweis: Die Erstellung eines virtuellen Fluges durch die 3D Szene mittels Animationskamera bietet oft die eindrucksvollsten Ansichten. Allerdings setzt die Definition des Pfades, sowie die Erstellung der Animation ein hohes Maß an Erfahrung in der Arbeit mit Autodesk 3D Studio MAX voraus. Der CityGRID® Builder bietet nur die Möglichkeit der Pfadwahl für Aug- und Zielpunkt an. Eingriffe in die Pfadgestaltung sowie den Animationsablauf müssen in Autodesk 3D Studio MAX selbst definiert werden und liegen nicht im Leistungsspektrum von CityGRID® Builder.



Hinweis: Kameraanimationen lassen sich auch direkt im CityGRID® Scout erstellen (vgl. Handbuch CityGRID® Scout)

Für weitere Informationen verweisen wir auf die Autodesk Dokumentation.

8 Export zum CityGRID® Scout

8.1 Steuerung des Exports zum CityGRID® Scout

Für jeden Eintrag, mit Ausnahme der vier Hauptknoten, kann im Projektextplorer festgelegt werden, ob die Daten, sowie alle unter dem selektierten Knoten befindliche Knoten in den CityGRID® Scout übernommen werden sollen. Dazu ist die Checkbox `Für CityGRID Builder verwenden` zu setzen. Wenn dieser Parameter nicht gesetzt wurde, verbleiben die entsprechenden Daten im 3D Studio Builder, werden aber nicht exportiert.



***Tipp:** Durch gezieltes Setzen des Parameters `Für CityGRID® Builder verwenden` können Konstruktionshilfen im 3D Studio eingesetzt werden, ohne dass diese vor dem Export der Daten für den Scout gelöscht werden müssen.*

Für Geometriedaten kann weiters die Datenklasse definiert werden. Diese steuert den Grad der Optimierung im Scout für die jeweilige Datengruppe. Zur Verfügung stehen folgende Klassen:


- **Gelände**
Umfasst alle 2,5D Daten die Gelände oder Oberflächenmodelle darstellen. Große Modelle sollten in ein regelmäßiges Raster unterteilt werden um die Optimierung günstig zu beeinflussen. Falls Texturen verwendet werden, sollte ein regelmäßiges, weitestgehend gleichmaschiges Mosaik von Bildern (z. B. Orthofotos) zum Einsatz kommen.
- **Gebäude**
Alle semantisch korrekten CityGRID® Gebäude fallen in diese Klasse. Gegenüber anderen 3D Daten zeichnen sich Objekte dieser Datenklasse dadurch aus, dass sie getrennt in Dach und Fassade (bzw. allfällige weitere Elemente) vorkommen und relativ kompakte Ausdehnungen haben. Bei der LoD Ableitung wird das Gebäude geometrisch immer mehr vereinfacht und kann im Extremfall auch komplett entfernt werden (z. B. kleine Objekte bei sehr großer Entfernung)
- **Entwurf**
Alle 3D Daten, die weder der Gelände noch der Gebäude Datenklasse zuzuweisen sind, finden sich hier. Im Gegensatz zu den beiden anderen Datenklassen können Daten vom Typ Entwurf den speziellen Optimierungsalgorithmus zur Aufbereitung der LoD Stufen nicht nutzen, wodurch bei sehr großen Szenen die Performance der Szene beeinträchtigt sein kann. Dafür ist wiederum sichergestellt, dass alle Daten, auch in niedrigen LoD Stufen, stets vorhanden sind und nie entfernt werden.




***Hinweis:** Im Allgemeinen wird die Datenklasse durch den 3D Studio Builder automatisch gesetzt und muss nicht verändert werden.*

8.2 Daten ins Builder-Projekt übernehmen

Der CityGRID® 3D Studio Builder ist gemeinsam mit dem FME Builder das optimale Werkzeug, um Szenen für den CityGRID® Scout vorzubereiten. Nachdem alle Szenen fertig gestellt sind und Kameras definiert wurden, können die Rohdaten für ein Scout-Projekt exportiert werden. Die eigentliche Erstellung des Scout-Projekts ist von der Rohdatenprozessierung entkoppelt und wird über das Builder Control Center gesteuert. Dadurch ist es möglich sukzessive ein komplexes Projekt zu erstellen und erst nach Abschluss aller Arbeiten einmal die Optimierung für den Scout durchführen zu lassen.

Der Export wird durch Klicken auf den Button `Daten ins Builder – Projekt übernehmen`  der Haupt-Toolleiste gestartet. Dabei werden die CityGRID® Scout-Einstellungen aller Objekte berücksichtigt (vgl. III 8.1). In diesem Prozess werden eine Reihe von Vor-Optimierungen an den Daten für den CityGRID® Scout durchgeführt. Der Prozess dauert je nach Datenmenge einige Minuten bis mehrere Stunden. Am Ende erscheint eine Mitteilung, dass der Prozess fertig ist.



Hinweis: Wenn die **SHIFT** (Umschalten) Taste beim Klick auf **CityGRID Scout Projekt erstellen**  gedrückt wurde, wird nach der Rohdatenaufbereitung sofort das **Builder Control Center** geöffnet.

CityGRID Scout Projekt erstellen  **+ALT** öffnet das Scout-Projekt falls dieses über das **Builder Control Center** schon einmal prozessiert wurde.

IV. Builder Control Center

Das Builder Control Center ist ein zentrales Verwaltungstool für Builder-Projekte. Mittels Builder Control Center können neue Builder-Projekte erstellt, zurückgesetzt oder gelöscht werden. Ebenso sind die Zusammenstellung von Szenen zu Varianten, sowie die Kombination mehrerer Builder-Projekte Aufgaben des Programms. Schließlich bietet das Builder Control Center noch die Möglichkeit die Datenoptimierung für CityGRID® Scout Projekte, und damit die eigentliche Erstellung des Projekts aus den vorprozessierten Rohdaten, zu starten und erstellte Scout-Projekte für die Verteilung freizuschalten.



Hinweis: Das Builder Control Center steht momentan nur in englischer Sprache zur Verfügung

1 Arten von Builder-Projekten

Über das Builder Control Center lassen sich unterschiedliche Arten von Builder-Projekten erstellen und verwalten. Es können drei Arten von Builder-Projekten definiert werden.

1.1 Einfaches Builder-Projekt

Das einfache Builder-Projekt entsteht durch die Rohdatenprozessierung in 3D Studio Max oder FME. Es zeichnet sich dadurch aus, dass die Rohdaten im Builder-Projekt physisch vorhanden sind. Die Rohdaten können in mehreren Zyklen durch FME und/oder 3D Studio Max erstellt werden und in Form von Szenen abgelegt sein.

Bei der Optimierung des Builder-Projekts für den CityGRID® Scout wird auf die Rohdaten zugegriffen und die Scout-Datenstruktur aufgebaut. Das Scout-Projekt kann anschließend freigeschalten und verteilt werden.

Einfache Builder-Projekte eignen sich für lokale Projekte, oder für die Aufnahme von Daten einer bestimmten Datenklasse (Gebäude, Gelände, Punktwolken etc.). Sie stellen die Basis für kombinierte Builder-Projekte dar, wie sie nachstehend beschrieben sind.

1.2 Managed-Builder-Projekt

Managed-Builder sind Vereinigungen von einfachen Builder-Projekten. Sie referenzieren die eingebundenen einfachen Builder-Projekte und greifen auf die darin gespeicherten Rohdaten zu. Die Rohdaten selbst verbleiben aber bei den eingebundenen einfachen Builder-Projekten. Während der Erstellung des Scouts (*Create* vgl. 5.1) werden aus den Rohdaten gemeinsame Datenbäume je Dateiklasse erstellt, die ein nachträgliches Separieren der Rohdaten verhindern. Ergebnis ist eine optimierte Datenhaltung, allerdings bedarf jede Änderung eines referenzierten Teilprojekts die komplette Neuberechnung des Scouts. Genauso wie bei einfachen Builder-Projekten lässt sich eine Variantenschaltung umsetzen.



Hinweis: Managed-Builder können zu keinen weiteren Managed-Builder-Projekten hinzugefügt werden, da diese über keine Rohdaten mehr verfügen. Diese sind bei der Optimierung des Scout-Projekts aber zwingend notwendig.

Um ein Managed-Builder-Projekt zu erstellen ist ein leeres Builder-Projekt anzulegen (vgl. IV 3) in das dann die gewünschten Builder-Projekte eingebunden werden (vgl. IV 4.1)

1.3 Vereinigte Builder-Projekte (Gemergte Builder-Projekte)

Vereinigte, oder gemergte Builder-Projekte referenzieren einfache oder Managed-Builder-Projekte und kombinieren darin befindliche Scout-Projekte miteinander, ohne diese einer erneuten Optimierung zuzuführen. Es erfolgt keine Optimierung der Scout-Projekte mehr, sondern nur noch die gemeinsame Darstellung der Teil-Scouts (*Merge* vgl. 5.2). Auf die Einzeldaten gibt es keinen Zugriff mehr und auch die Variantenschaltung kann in diesem Schritt nur noch auf die Datenpakete der Teil-Scouts durchgreifen. Jeder Teil-Scout erscheint mit seiner eingestellten Konfiguration und Variantenschaltung, die sich über die Seite *Config* (vgl. 5.3) noch beeinflussen lässt.

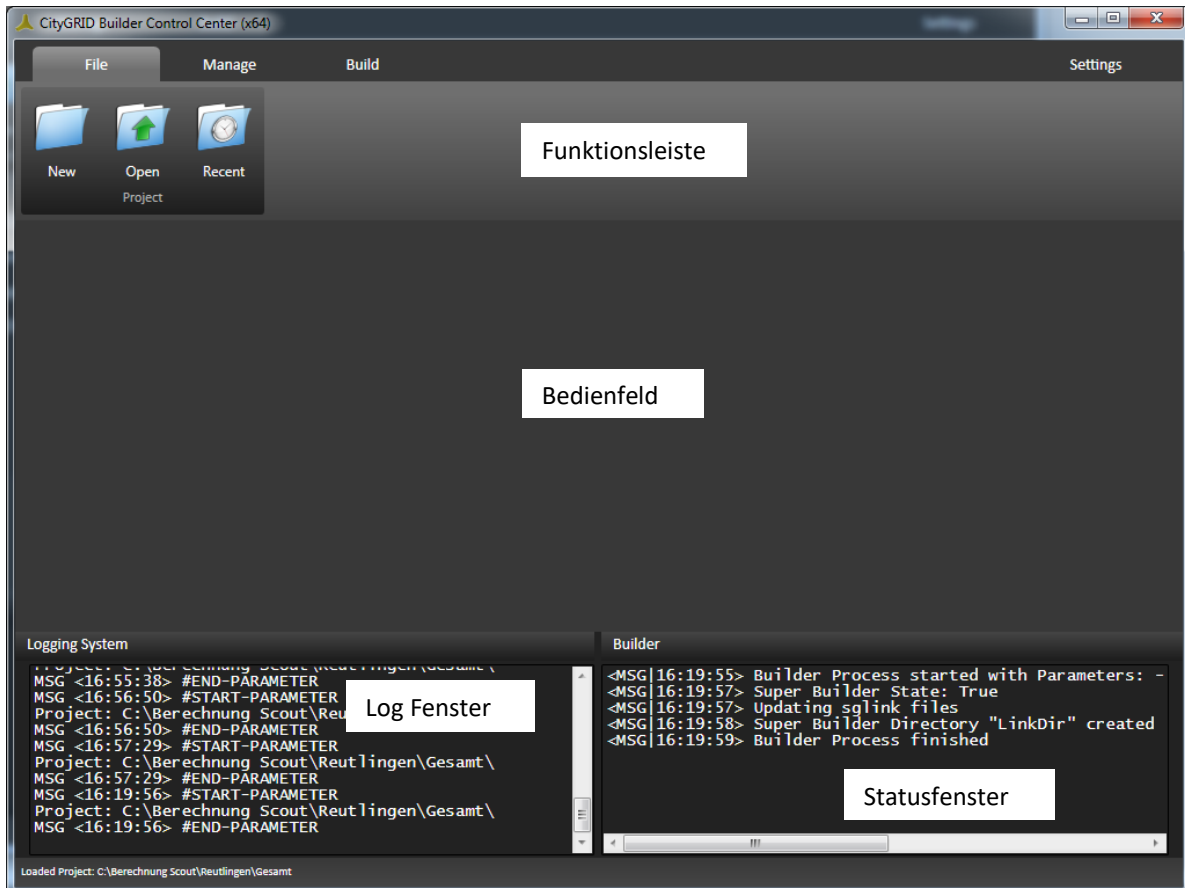
Solange ein gemergter Scout nicht publiziert wurde, greift der Scout auf die eingebundenen Teil-Scouts lokal zu. Erst beim Publizieren werden alle Teil-Scouts in das Scout-Verzeichnis des gemergten Builder-Projekts kopiert.




Hinweis: Gemergte Builder-Projekte können zu keinen weiteren Managed-Builder bzw. gemergten Builder-Projekten hinzugefügt werden.

Um ein gemergtes Builder-Projekt zu erstellen, ist ein neues Builder-Projekt anzulegen (vgl. IV 3) in das dann die gewünschten Builder-Projekte eingebunden werden (vgl. IV 4.1)

2 Die Builder Control Center Oberfläche

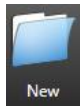


Die Oberfläche des Builder Control Center gliedert sich in folgende Bereiche:

- **Funktionsleiste**
In ihr sind die Programmfunktionen untergebracht. Über das Reitersystem kann auf unterschiedliche Funktionen zugegriffen werden.
 - **Bedienfeld**
Je nach gewählter Funktion der Funktionsleiste werden im Bedienfeld weitere Einstellungen vorgenommen oder Prozesse gestartet
 - **Log Datei**
Jeder gestartete Prozess protokolliert in die SuGu.log Datei des Builder-Projekts. Diese Log Datei wird in Echtzeit an das Log-Fenster geleitet.
-  **Hinweis:** Über den Reiter Settings kann die Log Datei mittels Button *Reset* jederzeit gelöscht werden.
- **Statusfenster**
Im Statusfenster werden Start- und Endzeiten von laufenden Builder-Prozessen protokolliert.

3 File

Im Reiter File können neue Builder-Projekte angelegt, oder bestehende geöffnet werden.



Wenn ein neues Builder-Projekt erstellt werden soll, kann dies durch klicken auf den Button **New** durchgeführt werden. Es öffnet sich ein Dateibrowser, in dem der Speicherort für die Builder-Ordnerstruktur (das Builder-Projekt) festgelegt wird. Durch diesen Vorgang werden alle übrigen Funktionen des Builder Control Center aktiv.



Um ein bestehendes Projekt zu öffnen wird auf **Open** geklickt und im Dateibrowser das Builder-Projekt gewählt. In diesem gibt es eine Datei *builder.sgp* (vgl. I 2), die mit dem CityGRID® Builder Control Center geöffnet werden kann. Wird diese Datei gewählt, öffnet sich das Builder-Projekt im Builder Control Center für alle angebotenen Funktionen.

Alternativ dazu kann auch ein bereits publiziertes Scoutproject (vgl. IV 5.4.3) geöffnet werden. Dazu ist der Dateityp auf „Scout-Projekt (sgp)“ zu ändern und die *project.sgp* oder *projectStarter.sgp* Datei zu wählen.



Hinweis: Geöffnete Scout-Projekte lassen sich nur nochmals publizieren (vgl. IV 5.4.3) alle anderen Funktionen des Builder Control Centers sind nicht verfügbar. Ein erneutes Publizieren von Scout-Projekten wird notwendig, wenn eine aktualisierte Version der Scout-Software mit den Daten verknüpft werden soll oder Einstellungen am Erscheinungsbild des Scouts einer Änderung unterworfen werden.



Beim Start des Builder Control Centers wird die Liste der zuletzt geöffneten Projekte (Builder und Scout) im Bedienfeld angezeigt. Der Button **Recent** lässt diese Liste bei jedem Klick erneut erscheinen. Ein Projekt kann durch Klick auf den Button **Open** oder über einen Doppelklick gestartet werden. **Remove** entfernt selektierte Projekte aus der Liste der zuletzt geöffneten Projekte.

4 Manage

Im Reiter Manage können Einstellungen zur Administration und Kombination von Builder-Projekten vorgenommen werden.

4.1 Projects



Um Scout-Projekte zu realisieren, müssen die dargestellten Daten in Builder-Projekten arrangiert und vorgehalten werden. Wenn die Projekte nicht zu groß werden, ist zumeist ein einfaches Builder-Projekt ausreichend. Bei umfangreichen Projekten macht es hingegen oftmals Sinn die Datenpakete auf unterschiedliche Builder-Projekte zu verteilen und für den Scout eine Kombination mehrerer Builder-Projekte zu betreiben.



Tipp: Die Verteilung auf mehrere Builder-Projekte kann etwa nach der Datenklasse (Gebäude, Gelände, Fremddaten), oder nach administrativen Einheiten (Blattschnittkacheln, Bezirksgrenzen etc.) erfolgen.

Über den **Projects** Button wird der Modus zum Kombinieren (linken) bestehender Builder-Projekte aktiviert.



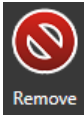
Hinweis: Wenn Builder-Projekte kombiniert werden so muss ein neues, leeres Builder-Projekt über den Button **New** am Reiter File angelegt werden. In dieses lassen sich dann die Verweise zu den bereits aufbereiteten Builder-Projekten, über den **Projects** Button einfügen.



Klickt man auf den Button **Add** öffnet sich ein Dateibrowser in dem ein vorhandenes Builder-Projekt eingebunden werden kann. Dazu ist im Builder-Projekt die Datei *builder.sgb* zu wählen. Für jedes so gewählte Builder-Projekt erscheint im Bedienfeld des Builder Control Centers eine Zeile mit dem Pfad zur *builder.sgb* Datei. Es können beliebig viele Builder-Projekte derart eingebunden werden.



Hinweis: Vereinigte (gemergete) Builder-Projekte können über den Projects Button nicht eingebunden werden. Ein gemergetes Builder-Projekt stellt die finale Ausbaustufe eines Builder-Projekts dar und lässt sich selbst nicht mehr weiterverarbeiten.



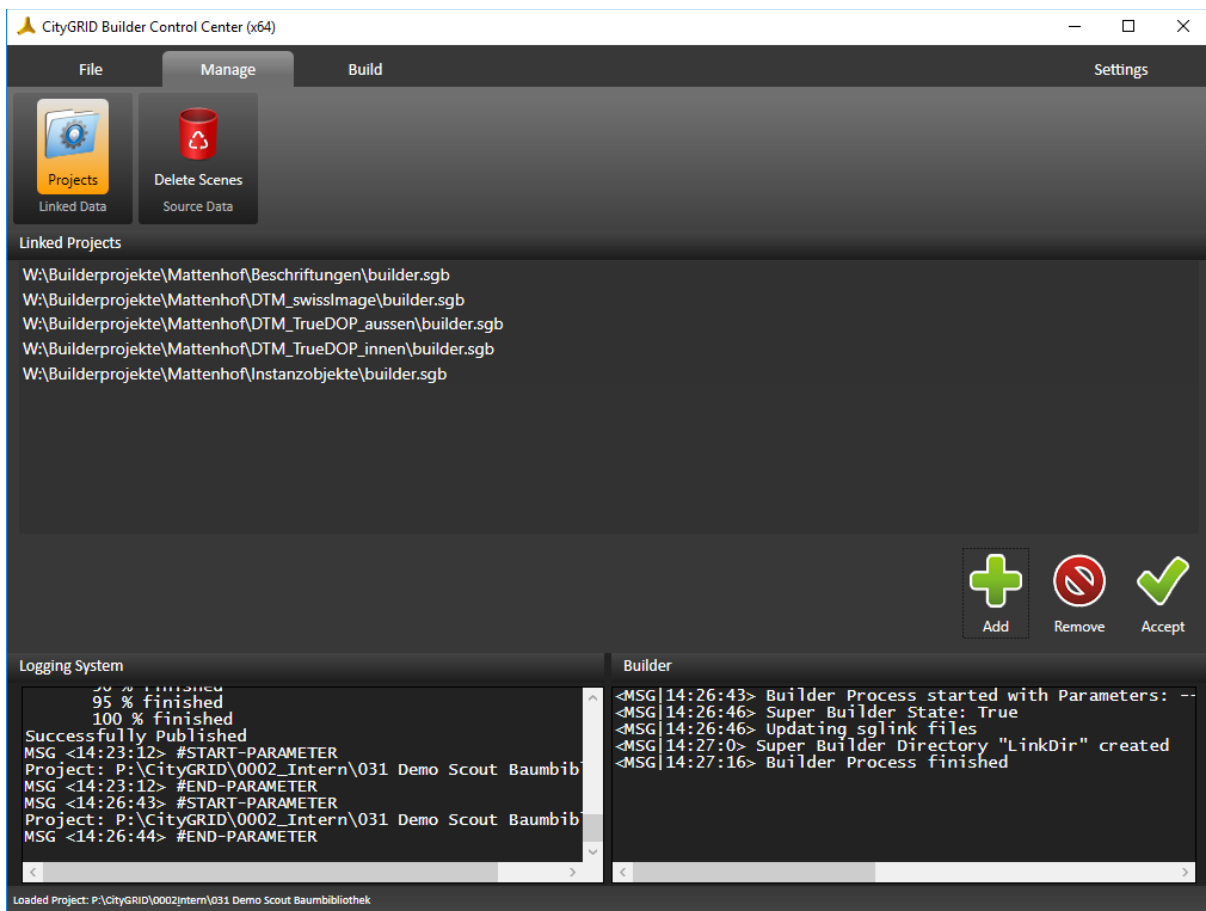
Unerwünschte Builder-Projekte werden im Bedienfeld des Builder Control Centers angewählt und lassen sich über den Button **Remove** entfernen.



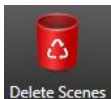
Nach Abschluss der Arbeit wird die getroffene Auswahl mittels Button **Accept** bestätigt



Tipp: Wechselt man den Reiter, bevor die Änderungen bestätigt wurden, gibt das Builder Control Center eine entsprechende Meldung an den Nutzer aus. Nutzen Sie diese Meldung um unerwünschte Änderungen rückgängig zu machen.

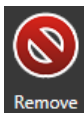


4.2 Delete Scenes

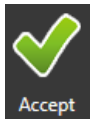


Über den Button **Delete Scenes** kann auf die angelegten Szenen des eingelesenen Builder-Projekts zugegriffen werden. Verwenden Sie die Funktion dazu nicht mehr benötigte Daten aus dem Builder-Projekt zu löschen.

Das Bedienfeld zeigt die Szenen nach den Datenklassen gruppiert an. Durch Setzen der Checkbox nach dem Pfad kann diese markiert werden. Optional lässt sich auch die Checkbox hinter der Datenklasse setzen. Dies hat zur Auswirkung, dass alle Szene der Datenklasse markiert/ entmarkiert werden.



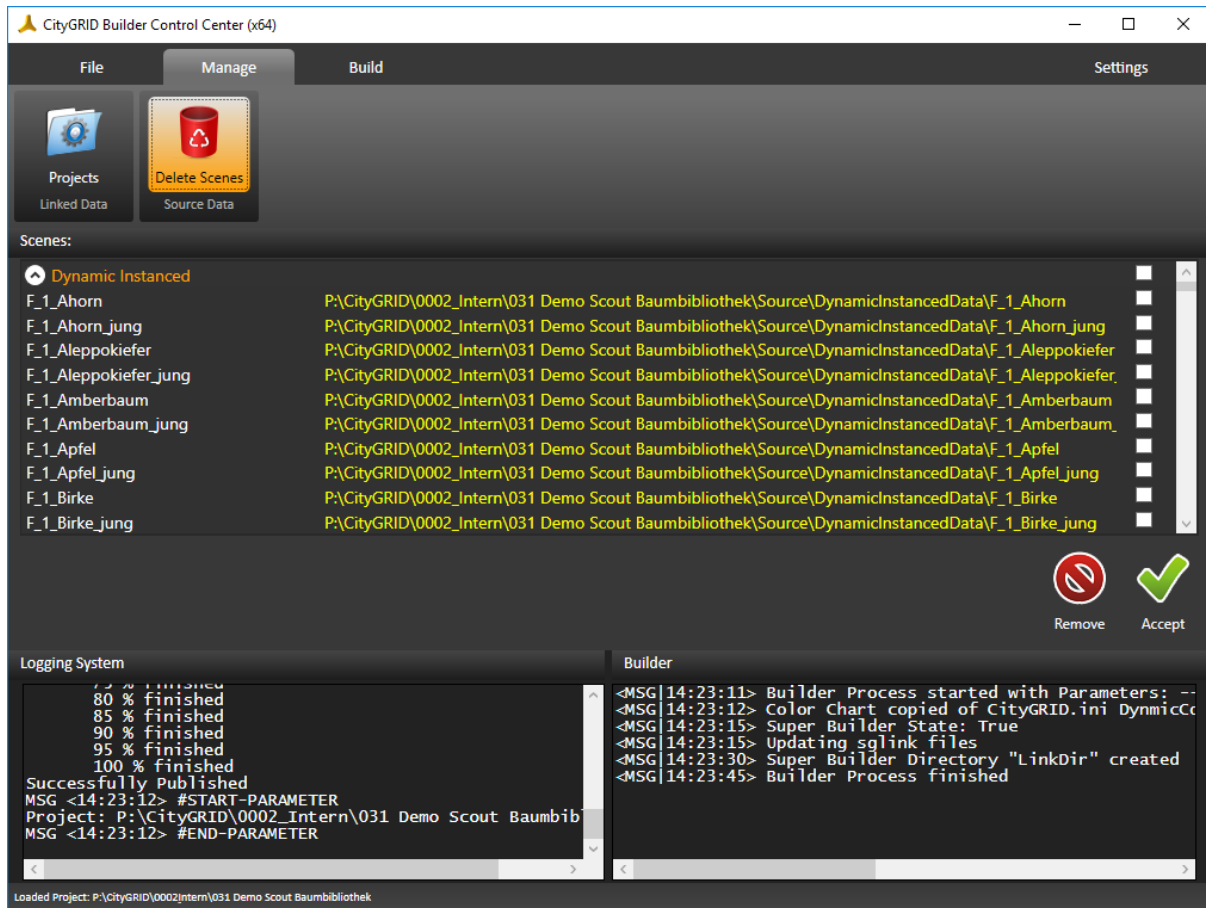
Durch Klick auf den Button **Remove** werden alle ausgewählten Szenen zum Löschen vorgemerkt und aus dem Bedienfeld gelöscht, an den Daten selbst erfolgt noch keine Änderung.



Erst beim Klick auf den Button **Accept** löscht das Builder Control Center die vorgemerkten Dateien im Builder-Projekt.



Hinweis: Diese Aktion kann nicht rückgängig gemacht werden. Gelöschte Daten werden physisch am Datenträger entfernt. Beim Klick auf **Accept** erscheint auch keine erneute Sicherheitsabfrage mehr.



5 Build

In diesem Tab wird die Aufbereitung des Scout-Projekts gestartet, sowie die Verteilung außerhalb des Lizenzsystems durchgeführt.

5.1 Create



Alle Szenedaten aus FME und 3D Studio Builder müssen vor dem Laden in den CityGRID® Scout einer Datenoptimierung und -restrukturierung unterzogen werden. Dabei werden die bisherigen Dateien aufgelöst, zu Varianten kombiniert, LoD Stufen von Geometrie und Bilder angelegt, Datenbäume (Voxel-Octrees) gebildet und performanceoptimierte Dateien neu erstellt. Die Steuerung dieser Vorgänge erfolgt über den Button **Create**.

5.1.1 Szenen und Varianten

Um Szenen in unterschiedlichen Kombinationen im CityGRID® Scout zeigen und zur Laufzeit wechseln zu können steht das Konzept der Varianten zur Verfügung. Jede Szene eines Builder-Projekts kann zu beliebig vielen Varianten zugeordnet werden und ein Builder-Projekt selbst kann beliebig viele Varianten besitzen.

Varianten regeln nur die Sichtbarkeit von Szenen während das Scout-Projekt läuft. Die Daten selbst werden nur einmal berechnet und bei Bedarf an die Grafikkarte gesandt.

Im Bedienfeld erscheinen alle Szenen die über den FME oder 3D Studio Builder im geladenen Builder-Projekt angelegt wurden. Die Szenen werden dabei nach Datenklassen gruppiert (vgl. II 2.2) und haben neben dem Szenenamen auch das Quell-Builder-Projekt (in gelber Schrift) angegeben.

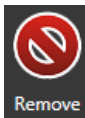


Hinweis: Im Falle eines Super-Builder erscheinen im Bedienfeld nun alle Szenen der eingebundenen Builder-Projekte.

Wenn bereits Varianten definiert wurden, werden diese ebenfalls eingelesen und die Zuordnung der Szenen zu den Varianten angezeigt. Falls keine Varianten in einem Projekt vorhanden sind, legt das Builder Control Center automatisch eine Variante „New Variant“ an.



Neue Varianten können durch Klick auf den Button **Add** erstellt werden. Durch Doppelklick auf den Variantennamen in der linken Spalte des Bedienfeldes kann ein beliebiger Name vergeben werden.



Nicht mehr benötigte Varianten lassen sich mittels Button **Remove** aus dem Builder-Projekt entfernen.



Hinweis: Varianten lassen sich auch aus den aufbereiteten Datenpaketen über die Config-Seite festlegen, und müssen nicht bereits während Create festgelegt werden. Dazu muss jedes Datenpaket, das in eine Variante eingebracht werden soll, individuell erstellt werden. Die Festlegung von Varianten bei der Kombination von Builderprojekten in einem merged Builder ist die empfohlene Vorgehensweise.

5.1.2 Prozessierungsmodi

Je nach Größe des Scout-Projekts stehen die zwei Optimierungsverfahren *Project* und *City* zur Verfügung, die unter der Rubrik *Properties* angeboten sind. Die Verfahren bereiten die Scout-Daten grundlegend unterschiedlich auf und stellen unterschiedliche Anforderungen an eingesetzte Softwaremodule und Rechenzeiten. Auch das Datenhandling im Scout ist grundverschieden.



Hinweis: Beide Prozessierungsmodi nutzen die GPU (Grafikprozessor) zum Aufbereiten der Texturbilder. Aus diesem Grund muss verhindert werden, dass die Grafikeinheit während der Berechnung inaktiv wird, oder gar nicht zur Verfügung steht. Insbesondere bei Aktivierung von Bildschirmschonern und dem Arbeiten über die Remote Desktop Verbindung von Windows treten diese Effekte auf. Achten Sie daher in den Windowseinstellungen darauf, dass die Grafikkarte nie deaktiviert wird und setzen Sie Drittsoftware (z.B. TeamViewer) bei Remotesteuerung des Builderrechners ein.

Prozessierungsmodus Project

Der Prozessierungsmodus *Project* ist für lokale, kleinräumige Projekte gedacht. Es werden alle Daten derart aufbereitet, dass diese komplett im Hauptspeicher bzw. Grafikspeicher gehalten werden können. Ein Nachladen von Daten erfolgt bei dieser Prozessierungsmethode nicht. Die Texturen werden in unterschiedlicher Auflösung vorberechnet (LoD) und die Geometrie in für den Scout optimierten Files abgelegt. Beim Start des Scout-Projekts ermittelt sich der Scout die verfügbare Speichermenge und wählt automatisch die mögliche Auflösung der Texturen. Die Geometrie wird stets mit der vollen Datenmenge angezeigt.



Hinweis: Auf Grund der einfacheren Datenaufbereitung ist die Rechenzeit deutlich kürzer als beim Prozessierungsmodus City. Dennoch können auch bei diesem Modus Rechenzeiten von mehreren Stunden auftreten.

Prozessierungsmodus City

Um großräumige Projekte (ganze Städte oder Regionen) mit hoher Datendichte realisieren zu können steht der Modus *City* zur Verfügung. Im Gegensatz zum *Project* Modus erfolgt die Organisation der Geometrie- und Texturdaten im Scout entfernungsabhängig. Zur Laufzeit des Scouts werden die Daten dynamisch nachgeladen, sodass im Nahbereich stets die bestmögliche Auflösung von Geometrie und Texturen zur Anzeige gebracht wird. Je weiter entfernt Daten vom aktuellen Fokus des Scout-Projekts sind, desto generalisierter werden diese dargestellt. Die Organisation der Geometrie-daten erfolgt nun in Form von Datenbäumen, in denen neben den nicht reduzierten Daten auch die generalisierten Zwischenstufen (LoDs) gespeichert sind. Zum Aufbau der Zwischenstufen wird Funktionalität von 3D Studio Max eingesetzt, weshalb eine Instanz, samt verfügbarer Lizenz, von 3D Studio Max am Computer verfügbar sein muss. Falls kein 3D Studio Max verfügbar ist, kann eine

vereinfachte Baumstruktur ohne die LoD Stufen berechnet werden. Derartig erstellte Projekte sind in der Sichtweite aber deutlich beschränkt.



Hinweis: Bei dieser Berechnungsmethode ist mit hohen Rechenzeiten und erheblichem Datenvolumen zu rechnen. Je nach Datenmenge sind mehrere Stunden bis Tage Rechenzeit normal.



Hinweis: Der Prozessierungsmodus „City“ setzt eine Installation des CityGRID® Autodesk Plugins (CityGRID® 3D Studio Max Builder), sowie eine FME Installation voraus. Fehlt eines der genannten Programme, bzw. die entsprechende CityGRID® Installation erscheint ein Warnungsfenster.

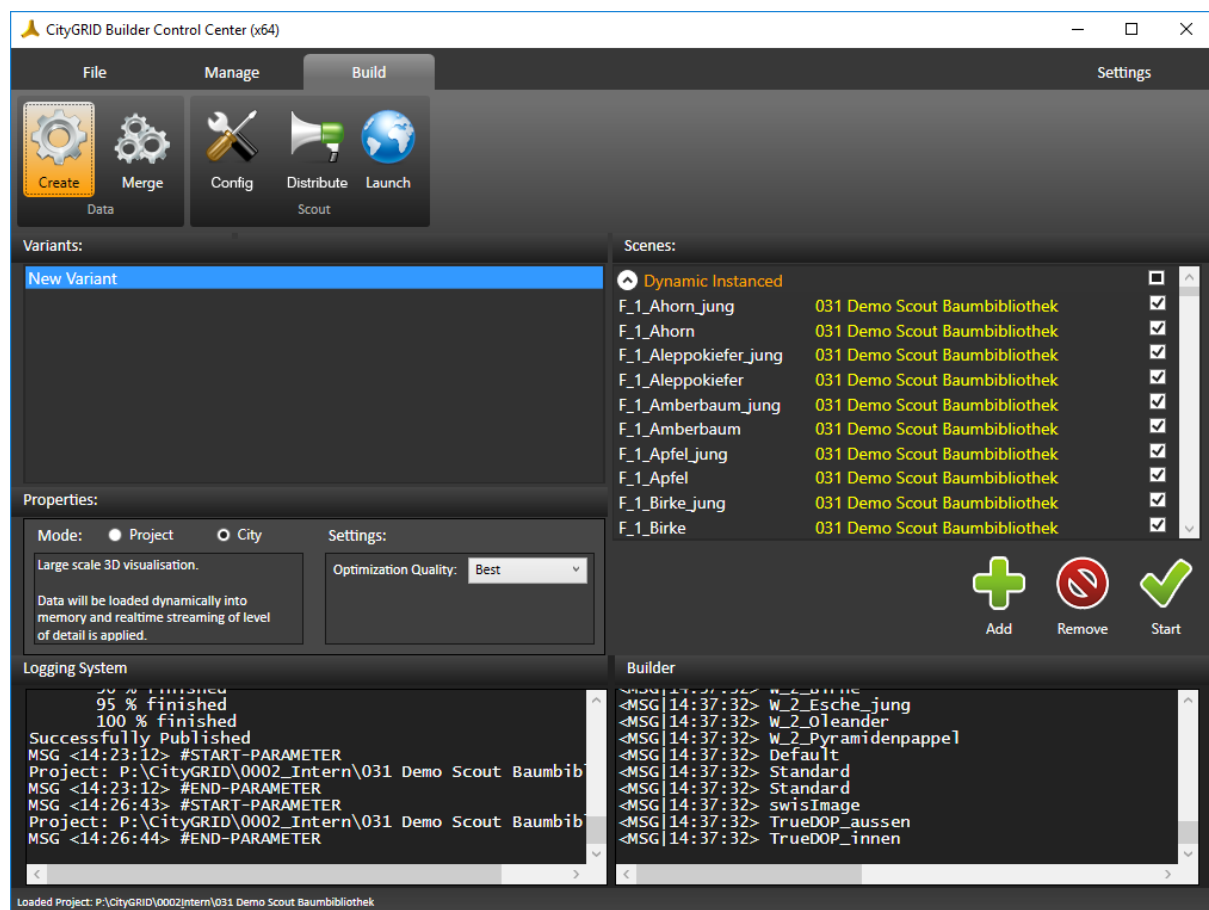
5.1.3 Settings

Bei der Optimierung der Scout-Daten müssen Texturen in eine optimierte Datenstruktur überführt werden. Je nach Güte der verwendeten Daten kann diese Optimierung stärker oder schwächer angewandt werden, was unmittelbare Auswirkung auf die Berechnungszeit sowie die Qualität der Darstellung im Scout hat.

Neben der Optimierungsstärke kann auch die Integration von allfällig vorhandenen Attributdaten bei CityGRID® Datensätzen vorgenommen werden.

Optimization Quality

Dieser Wert legt fest wie gut die Optimierung der Datenbäume, insbesondere hinsichtlich der Texturoptimierung, stattfinden kann. Maßgeblich dafür ist die sinnvolle Zuweisung von Texturen zu Flächen. Wenn Texturen auf mehrere Flächen unterschiedlicher Ausdehnung und Ausrichtung gelegt werden kann die Pixelauflösung des Bildes nicht korrekt berechnet werden. Die Folge davon ist, dass die Texturen im Scout grob pixelig bzw. stark verwaschen wirken. In diesen Fällen muss die Optimierungsqualität sukzessive reduziert werden um visuell ansprechende Ergebnisse im Scout erzielen zu können.





***Tipp:** Mit CityGRID® texturierte Daten können im Allgemeinen mit der besten Optimierung verarbeitet werden. Bei Fremddaten hängt es von der Art der Erstellung ab, ob Abstriche bei der Optimierung zu machen sind. Fremddaten sollten daher in eigenen Scout-Projekten geführt werden, die im Bedarfsfall einer neuen Optimierung unterworfen werden können.*

5.1.4 Durchführung der Optimierung

Um Szenen einer Variante zuzuordnen muss folgendermaßen vorgegangen werden:

1. Eine neue Variante anlegen oder eine bestehende aktivieren.
2. In der Spalte *Scenes* die Checkbox der zuzuordnenden Szenen setzen.



***Tipp:** Falls alle Szenen einer Datenklasse (Static Building, Dynamic Design etc.) zugewiesen werden sollen, kann die Checkbox neben dem Klassennamen gesetzt werden. Falls innerhalb einer Datenklasse nicht alle Szenen gesetzt sind, wird anstatt des Hakens in der Checkbox ein Quadrat gezeichnet)*

3. Vorgang für alle Varianten wiederholen
4. Den gewünschten Prozessierungsmodus unter *Properties* einstellen.



5. Optimierung des Scout-Projekts durch Klick auf **Start** einleiten.



***Hinweis:** Sobald **Start** geklickt wird, werden die Änderungen durchgeführt und gleich mit dem Builder-Projekt gespeichert. Eine aktive Optimierung kann im Builder Control Center nicht abgebrochen werden*



Nach erfolgreicher Optimierung der Daten entsteht am Ende dieses Vorgangs ein Scout-Projekt, das mittels CityGRID® Scout betrachtet werden kann. Der Button **Launch** wird in diesem Fall aktiv, ebenso der Button **Distribute**. Wird auf **Launch** geklickt öffnet sich das soeben berechnete Projekt, vorausgesetzt eine entsprechende Scout-Lizenz ist am Rechner verfügbar.



***Hinweis:** Sollte die Erstellung auf ein Problem auflaufen findet sich im Log Fenster die entsprechende Logdatei mit einer genauen Fehlerbeschreibung. Senden Sie diese an support@uvmsystems.com gemeinsam mit einer Beschreibung der Entstehungsgeschichte des Builder-Projekts.*

5.2 Merge (Vereinen berechneter Scouts)

Große Scout-Projekte können effizient erstellt werden, in dem zuvor Teil-Scouts gebildet und anschließend gemeinsam zur Anzeige gebracht werden. Diesen Ansatz verfolgt der gemergete Builder bzw. gemergete Scout. Dabei werden zuvor optimierte Scouts einem neuen Builder-Projekt zugewiesen und über die Erstellungsmethode **Merge** vereint. Der Vorteil dieser Vorgehensweise liegt darin, dass jeder Teil-Scout für sich optimiert und inspiziert werden kann. Erst wenn diese Teilprojekte zufriedenstellen erstellt wurden, erfolgt die Vereinigung, wobei eine erneute Optimierung unterbleiben kann.



***Tipp:** Wenn in einem gemergeten Scout Varianten angezeigt werden sollen, so sind dafür einfache Builder oder managed-Builder-Projekte mit berechneten Scouts anzulegen und über die Config Seite (vgl. 5.3) zu kombinieren.*

Ein gemergtes Builder-Projekt stellt aktuell den Endzustand eines Builder-Projekts dar. Derartige Builder-Projekte können nicht weiter zusammengefasst werden.


5.2.1 Erstellung eines vereinten Scouts

1. Am Reiter *File* ist mittels Button  ein neues Builder-Projekt anzulegen.
2. In dieses werden am Reiter *Manage* über  einfache Builder-Projekte oder Super-Builder mit berechneten Scouts (vgl. IV 5.1) hinzugefügt.
3. Am Reiter *Build* kann nun über den Button  die Kombination der zuvor gelinkten Daten erfolgen.



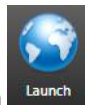
Hinweis: Falls eines oder mehrere der gelinkten Builder-Projekte keinen gültigen Scout aufweist, erscheint eine entsprechende Fehlermeldung und die Vereinigung wird nicht durchgeführt.

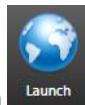


4. Durch Klick auf  beginnt die Vereinigung. Dabei wird vorerst in der Datei *ProjectStarter.sgp* des Scout-Ordners der Pfad zu jedem gelinkten Scout-Projekt eingetragen. An den Daten der gelinkten Scouts erfolgt keine Veränderung.



Hinweis: Das Vereinen von Scouts ist in wenigen Augenblicken erledigt da keine umfangreichen Optimierungsarbeiten mehr durchgeführt werden müssen. Allerdings lässt sich der Scout nur starten, wenn auf die referenzierten Daten auch zugegriffen werden kann. Zur Verteilung eines solchen Scouts müssen die referenzierten Scoutdaten in das Scout-Verzeichnis des merged Builder-Projekts kopiert werden, was über die *Distribute* Seite (vgl. 5.4) möglich ist.

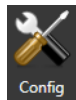


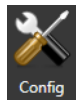
Der vereinte Scout kann über den Button  gestartet werden.



Hinweis: Da beim Vereinen vorerst keine Daten neu entstehen sind gemergete Scouts nur am Erstellungsrechner nutzbar. Erst durch das Publizieren des Scouts erfolgt das Kopieren der Bestandteile ins Builder-Projekt (vgl. IV 5.4.3).

5.3 Config

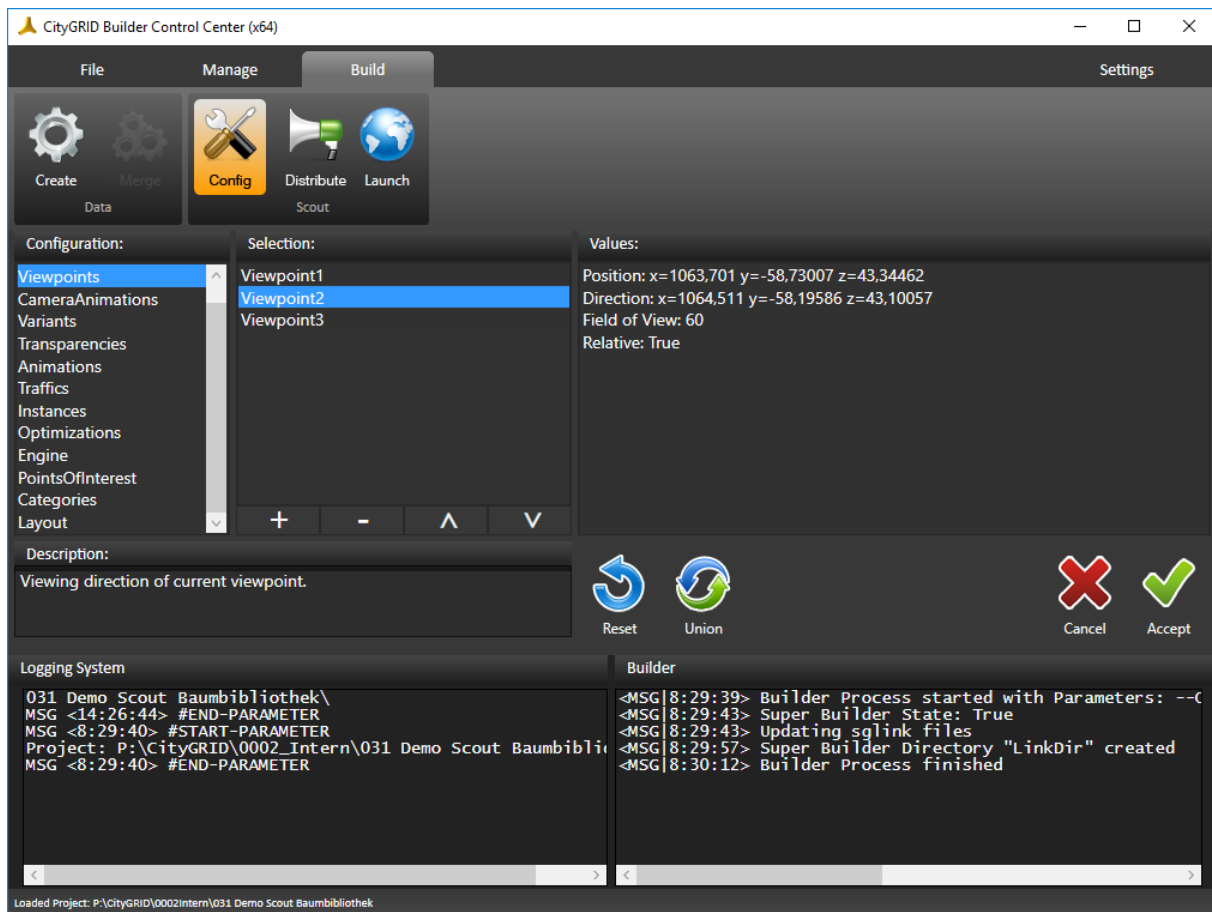


Der Button  ermöglicht den Zugriff auf diverse Einstellungen, die das finale Erscheinungsbild des Scouts festlegen. Beispielsweise lassen sich Blickpunkte, Varianten oder generelle Starteinstellungen des Scouts auf dieser Seite definieren.

Die *Config* Seite ist aus den Listen *Configuration*, *Selection* und *Values* aufgebaut, sowie dem Textfeld *Description* mit Beschreibungstext zum jeweiligen Parameter und der Buttons-Leiste. Die Funktionsweise ist für jeden Parameter gleich aufgebaut. In der Spalte *Configuration* wählt man die Rubrik aus, für die man Parameter setzen möchte. Sobald eine Auswahl für *Configuration* getroffen wurde, erscheinen unter *Selection* (vor-) definierte Parametersätze. Eine Auswahl unter *Selection* zeigt schließlich im Feld *Values* die zu bestimmenden Werte an.



Hinweis: Nicht jeder Eintrag unter *Configuration* besitzt Parametersätze die unter *Selection* gelistet werden. Manche Rubriken legen direkt die Werte unter *Values* offen.



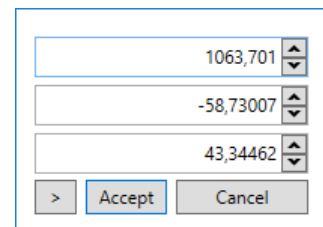
5.3.1 Parameter setzen

Das Setzen von Parametern über Config erfolgt stets nach demselben Schema:

1. Unter *Configuration* die gewünschte Kategorie auswählen. Daraufhin erscheinen in der Liste *Selection* die verfügbaren Parameter, bzw. werden die Buttons für das Erstellen/Editieren aktiviert.
2. Unter *Selection* nun den gewünschten Parametersatz auswählen um die editierbaren Werte unter *Values* angezeigt zu bekommen. Falls der Auswahl unter *Configuration* keine Parametersätze zugeordnet werden können, kann direkt mit dem Editieren der Werte unter *Values* begonnen werden.

Erlaubt die Auswahl aber das Definieren von Parametersätzen so können mittels Button **+** neue erstellt, bzw. über **-** existierende entfernt werden. Die Reihenfolge der Parametersätze lassen sich über die Buttons **^** **v** ändern.

3. Unter *Values* sind nun die entsprechenden Werte zu setzen. Dies können entweder Attributwerte sein, oder aber auch Datenpakete um etwa Varianten oder Transparenzen festzulegen. Jeder Eintrag der Liste *Values* lässt sich mittels Doppelklick editieren. Abhängig von der Art des Parameters sind nun unterschiedliche Eingaben möglich. Die Palette reicht hier von einer einfachen Texteingabe, über die Auswahl aus vordefinierten Listen, bis hin zur Übernahme von Werten direkt aus einem laufenden CityGRID® Scout.



Die Übernahme von Werten aus einem laufenden Scout erfolgt mittels Button **>** und ist insbesondere für die Übernahme von Positionen und Blickrichtungen gedacht.



Hinweis: Damit das Builder Control Center mit dem Scout kommunizieren kann, ist im Scout die Option „Kommunikation“ im Menü Einstellungen → Anwendung zu aktivieren.



Hinweis: Bei Eingaben mit Dezimalstellen ist als Trennzeichen stets das Komma des eingestellten Tastaturlayouts zu verwenden.

Jedem Wert ist im Textfeld *Description* ein Beschreibungstext zugeordnet, der Auskunft über die Funktion, bzw. die erwartete Eingabe liefert.

5.3.2 Verfügbare Parameter

- **Geodetics:**
 - Um eine Übersichtskarte nutzen zu können ist die Angabe eines *EPSG-Codes* (<http://www.epsg-registry.org/>) der Daten im Scout notwendig. Es sind die Ziffern des Codes anzuführen.
 - Falls ein lokales Koordinatensystem verwendet wird, das keinem EPSG Code zugeordnet ist, lassen sich unter *Custom Offset* die Verschiebungswerte angeben, die zu den lokalen Koordinaten addiert werden müssen, um zu den EPSG Koordinaten zu kommen.
 - *Offline Map* steuert über die Werte *true* und *false* ob das Scout-Projekt nach Daten für eine Offline-Übersichtsmap durchsucht werden soll. Falls solche Daten gefunden werden, nutzt das Übersichtsfenster diese anstatt des eingestellten Online Web Map Services. Zu beachten ist, dass die Nutzung der Offline Map weiterhin eine aufrechte Internetverbindung voraussetzt, um die Platzierung der aktuellen Position im 3D Projekt in der Übersichtskarte darstellen zu können.
 - Für den Schattenwurf ist die Lage der Sonnenposition im Scout in geografischen Koordinaten (*Longitude* und *Latitude*) vorzunehmen. Es werden maximal drei Nachkommastellen berücksichtigt.

- **Viewpoints:**

Ist die Liste aller Viewpoints in allen (Teil-)Scouts des zu (gemergten) Scouts. Die Namen der Viewpoints lassen sich in der Liste *Selection* mittels Doppelklick verändern. Mit einem Doppelklick auf die Parameterwerte eines selektierten Viewpoints, können auch diese verändert werden (Position, Direction, Field of View)



Tipp: bei einem geöffneten zugehörigen Scoutprojekt und dort erlaubter Externer Kommunikation Sh. Scout Handbuch Kap.5.4.4 Können Koordinaten der dort eingestellten aktuellen Kameraposition mit dem Button > im durch Doppelklick auf die Werte geöffneten Dialog übernommen werden

- **CameraAnimations:**

Zeigt alle Kamerafahrten des Scouts an. Die Namen der Animationen lassen sich in der Liste *Selection* mittels Doppelklick verändern.



Tipp: Animationen stammen aus 3D Studio Max oder werden direkt im Scout erstellt (siehe Handbuch CityGRID® Scout)

- **Variants:**

Die Rubrik *Variants* zeigt alle Varianten aller (Teil-) Scouts, sowie die dahinterliegenden Datenpakete an und bietet die Möglichkeit auf die Variantenschaltung nach dem Optimieren des Scouts Einfluss zu nehmen.



Tipp: Die nachträgliche Variantendefinition ist bei vereinten (gemergten) Scouts (vgl. IV 5.3) notwendig.

In der Liste *Selection* lassen sich vorhandene Varianten mittels Doppelklick umbenennen. Über den Button wird eine neue Variante angelegt, entfernt die selektierte.

Die Liste *Values* zeigt alle Datenpakete, gruppiert nach den eingebundenen (Teil-) Scouts an. Durch Setzen der Checkboxen können diese zu neuen Varianten kombiniert oder bestehende aufgelöst werden.



Hinweis: Ein Datenpaket ist bei der Optimierung eines Scouts (vgl. IV 5.1) entstanden und stellt die Summe aller Szenen dar, die beim Create einer Variante zugeordnet wurden.

Für die Zuordnung eines Teil-Scouts ist der Name des zugehörigen Bilderprojekts angeführt.



Tipp: Idealerweise sind je Teil-Scout nur Daten eines Themas und einer Datenklasse enthalten. Somit ist über die Bezeichnung des Builderprojekts eindeutig erkennbar, welches Datenpaket einer Variante zugeordnet wird.

Wenn ein Datenpaket keiner Variante explizit zugeordnet wurde, erscheint es in der Liste *Values* mit dem Text *not specifically assigned* und dem Datentyp *Root* oder *Scene*. Andernfalls ist der Name der jeweiligen Variante des Scouts, bzw. sein Datentyp (Instance, Animation, Traffic, etc.) angeführt. Um ein Datenpaket einer Variante zuzuordnen zu können ist die entsprechende Checkbox zu setzen.

- **Transparencies:**

Bietet wie *Variants*, die Möglichkeit einzelne Datenpaketen transparent zu schalten. Transparente Bereiche lassen sich zur Laufzeit eines Scouts ein- und ausblenden (siehe Handbuch CityGRID® Scout)



Hinweis: *Transparencies* hat keinen Einfluss auf Materialien mit Alphakanal. Die Transparenzwerte der Texturen bleiben unverändert.

- **Animations:**

Liste die Definition aller bewegten Objekte eines (Teil-) Scouts auf. Die Animationen werden im 3D Studio Max definiert.



Hinweis: Wenden Sie sich bitte an Ihren Betreuer bei UVM Systems um Animationen im Scout umsetzen zu können.

- **Traffics:**

Liste alle Verkehrsobjekte des (Teil-) Scouts auf. Ist aktuell noch nicht implementiert.

- **Instances:**

Erlaubt die Höhenpositionierung von Instanzdaten. Analog zu den *Variants* sind auch hier unter *Values* alle Datenpakete der (Teil-)Scouts gelistet, die für die gewünschte 3D Positionierung genutzt werden können. Durch Setzen einer Checkbox wird das Datenpaket mit den Instanzobjekten verknüpft. Klick man auf den *Move to* Button ermittelt jedes Instanzobjekt für seine 2D Position den höchsten Schnittpunkt aller Datenpakete und platziert sich darauf.



Hinweis: Voraussetzung für die Höhenpositionierung von Instanzobjekten ist entweder ein publizierter Scout (vgl. 5.4.3) oder ein Managed-Builder (vgl. 1.2) in dem Instanzdaten und Flächendaten gemeinsam vorgehalten sind.

- **Optimizations:**

Erlaubt das Setzen der Sichtweite im Scout über den automatisch ermittelten Standardwert. Gültige Werte sind -10 bis 10. Diese Werte entsprechen den voreingestellten Entfernungsstufen, die auch über das Menü *Einstellungen* → *Leistung* des Scouts gesetzt werden können. Je geringer der Wert ist, desto schneller lädt die Szene. Im Gegenzug ist die Sichtweite aber entsprechend limitiert.



Hinweis: Die gewünschten Werte sind bei laufendem Scout über das Menü → *Einstellungen* → *Leistung* zu bestimmen. Zu beachten ist, dass eine erhöhte Sichtweite signifikant größere Anforderungen an den Computer stellt, sodass sichergestellt werden muss, dass die definierten Werte auch verarbeitbar bleiben. Bei einem Überschreiten der Leistungsgrenze finden sich im SuGu.log des Scouts entsprechende Einträge.

- **Engine:**

Stellt alle allgemeinen Parameter der Unity 3D Engine, wie Hintergrundfarbe, Beidseitiges Rendern, Inaktivitätscountdown, Punktgröße bei Punktwolken etc. dar

- *PointsOfInterest:*

Nachdem unter *Categories* eine oder mehrere Kategorien erstellt wurden, können hier POIs erstellt werden. Unter *Values* eines selektierten POIs sind folgende Werte essentiell für das Erstellen:

- ❖ GlobalPosition: Position des POI Popups, gewählt werden, kann zwischen 10 Werten. Default-Werte sind ausreichend
- ❖ GlobalDuration: Dauer der POI-Popupanzeige in Sekunden
- ❖ Global Offset: Hier kann die Position des POI-Popups verändert werden, relativ zu der angegebenen Position in Pixel.
- ❖ Category: Hier wird die übergeordnete Kategorie des POIs eingestellt
- ❖ Position: Hier wird die Position des POIs eingestellt. Wenn im aktuell geöffneten, zugehörigen Scoutprojekt die externe Kommunikation erlaubt wurde und die aktuelle Kameraposition der gewünschten POI-Position entspricht, kann mit dem Button > im Koordinatendialog die Scoutposition übernommen werden. Die POI-Position ist das Zentrum einer Boundingbox
- ❖ Visible Delta: Hier werden die Maße der Boundingbox angegeben (in Meter), in der der POI im Scout angezeigt werden soll. Außerhalb dieser Boundingbox wird der POI nicht angezeigt.
- ❖ Is Relative: Üblicherweise handelt es sich um relative Koordinaten in Scoutprojekten, kann hier jedoch umgestellt werden.
- ❖ Viewpoint Position: Hier werden die Koordinaten eines Viewpoints angegeben, der idealerweise im Vorfeld erstellt wurde und gleich benannt wurde wie der entsprechende POI. Bei diesem Viewpoint handelt es sich um die Ansicht, auf die gesprungen wird, wenn im Scout der POI ausgewählt wird.
- ❖ Viewpoint Direction: Hier wird die Blickrichtung des POI Viewpoints angegeben.
- ❖ Viewpoint Field of View: Hier wird der Blickwinkel des POI Viewpoints in Grad angegeben.
- ❖ Viewpoint Relative: Sind die Koordinaten des Viewpoints relativ?

Wenn die Einstellungen mit *Accept* übernommen wurden, muss der geöffnete Scout geschlossen und wieder geöffnet werden, damit die Änderungen wirksam werden.

In das POI-Popup kann ein aussagekräftiges Bild übernommen werden. Im Dateisystem des Scoutprojektes unter ScoutName/Content/Data/Images/POIs können Bilder im .jpg Format erstellt werden, die den gleichen Namen bekommen müssen, wie der POI selber. Empfohlen sind Bildgrößen nicht größer als 1024*640 px.

- *Categories:*

To create Points of Interests, categories must be defined beforehand, to which the POIs can be assigned later. As an example, shopping/restaurants/bars or similar. With a double click on *Color* at the values of a selected category, each category can be assigned a specific color.

- *Layout:*

Definiert Logos, bzw. die Beschriftung des Scouts.

- *Project:*

Erlaubt das Festlegen von Parametern, die für den gesamten Teil-Scout gelten

- *Texture Mode:*

bietet die Möglichkeit auf die Art der Texturkachelung Einfluss zu nehmen. Es stehen die Methoden Repeat, Clamp, Mirror und Mirror Once zur Verfügung. Repeat weist die 3D Engine an die Textur wiederholend aufzubringen. Diese Einstellung ist bei Daten aus dem 3D Studio Builder anzuwenden, die mit gekachelten Texturen erstellt wurden. Clamp bildet das Texturbild

einmalig auf der 3D Fläche ab. Dieser Modus ist bei Geländemodellen, die mittels Orthofotos texturiert wurden, sowie bei CityGRID® Gebäuden mit automatischer Texturierung anzuwenden. Mirror spiegelt die Texturen an einer Achse. Dieser Texturmodus ist nur in Ausnahmefällen anzuwenden.

- **Cull Mode:**

legt die Sichtbarkeit von Flächen im 3D Fenster fest. Standardmäßig sind in 3D Fenstern nur jene Flächen sichtbar, deren Flächennormale entgegen der Blickrichtung des Betrachters orientiert sind. Zur Auswahl stehen die Optionen System, off, front und back. System bezieht die Einstellungen vom Parameter „Double side“ der Configuration Engine, Off zeigt stets alle Flächen beidseitig an, front legt die Sichtbarkeit fix auf Flächen fest, deren Normale entgegen der Blickrichtung weist, Back wiederum zeigt stets nur Flächen an, deren Flächennormale in Blickrichtung weist.



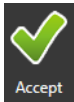
Hinweis: Bei den Cull Modi off, front und back hat die Einstellung Doubleside keinen Einfluss mehr und zur Laufzeit des Scouts lässt sich diese auch nicht mehr über die Taste F7 wechseln.



Reset verwirft alle Einstellungen und liest aus allen Teil-Scouts die Einstellungen neu aus. Sämtliche Einstellungen die rein im Builder Control Center gesetzt wurde, gehen durch diese Maßnahme verloren.



Union kombiniert alle aktuellen Einstellungen mit jenen der Teil-Scouts. Durch diese Maßnahme kann es vorkommen, dass Einträge wie etwa Viewpoints, doppelt vorkommen, da kein Angleich mit bereits bestehenden Einträgen geschieht. Unerwünschte Einträge müssen gegebenenfalls manuell entfernt werden.



Accept übernimmt alle Änderungen und speichert diese sofort beim Scout-Projekt ab. Dies gilt sowohl für bereits publizierte als auch noch unveröffentlichte Scouts gleichermaßen. Die *Config* Seite wird geschlossen.



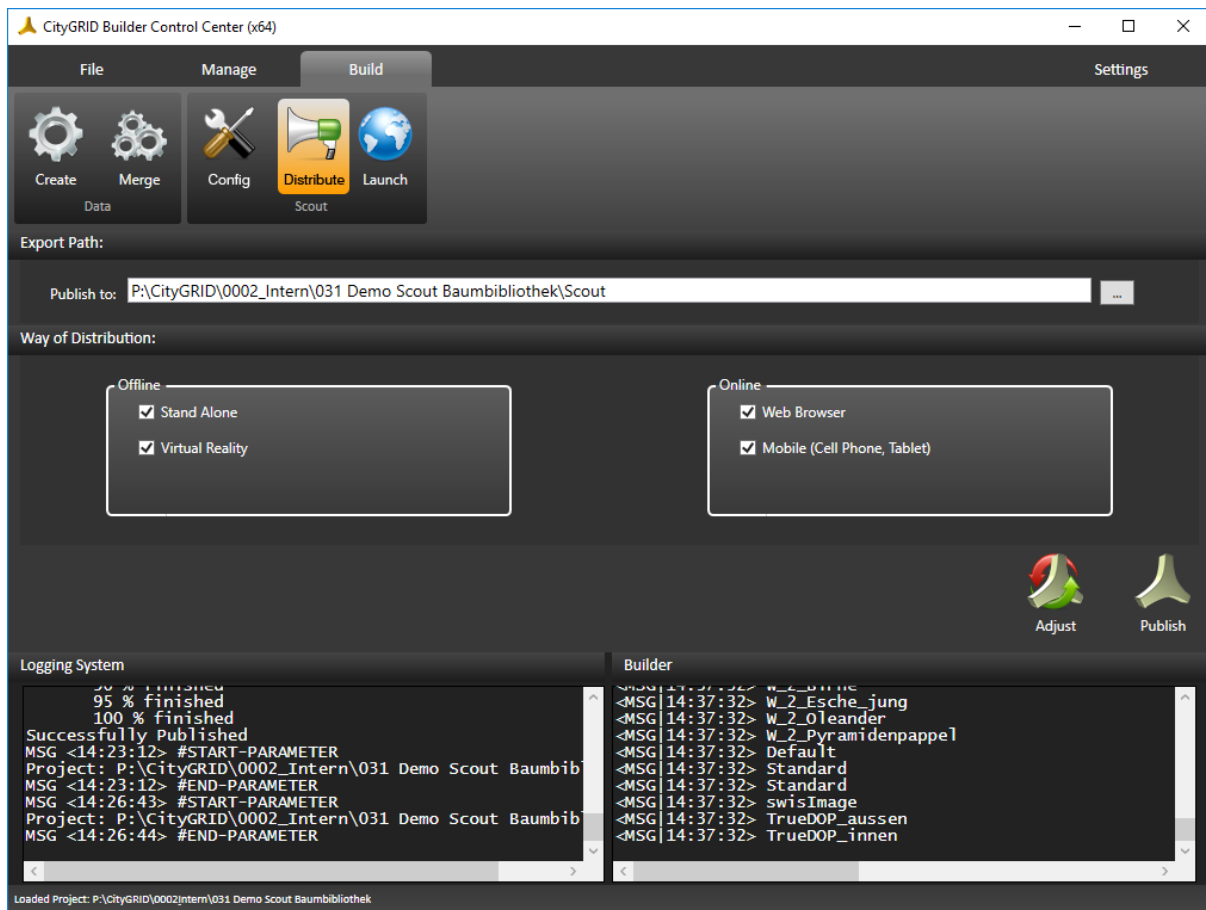
Cancel verwirft alle Änderungen und schließt die *Config* Seite ohne den Scout zu verändern.

5.4 Distribute


Korrekt berechnete Scout-Projekte werden standardmäßig mit der installierten Version des CityGRID® Scouts angezeigt. Diese Version setzt eine Verbindung zum CityGRID® Lizenzserver voraus und ist daher für die Verteilung der 3D Szene ungeeignet und dient nur der Kontrolle. Um das Scout-Projekt einem breiteren Anwenderkreis zugänglich zu machen muss eine Freischaltung erfolgen. Durch diesen Schritt erhält der Scout alle notwendigen Viewerdaten um losgelöst von jeder CityGRID® Installation lauffähig zu sein.



Hinweis: Voraussetzung für eine Freischaltung ist eine entsprechende Lizenz. Wenden Sie sich für nähere Auskünfte bitte an Ihren Betreuer bei UVM Systems GmbH.



5.4.1 Export Path

Bei der Freischaltung eines Scout-Projekts müssen die Scout-Daten in einem Builder-Projekt einer finalen Aufbereitung zugeführt werden. So kann etwa ein Kopieren von Teil-Scouts notwendig sein, oder die Aufbereitung des Bildmaterials für die Onlineverteilung. In jedem Fall befüllt sich das Scout-Verzeichnis des Builder-Projekts mit den finalen Daten. Über den Parameter *Publish to* kann der Speicherort dieses Arbeitsschritts festgelegt werden. Der Button  öffnet einen Dateibrowser in dem der Speicherort angegeben wird. Die aufbereiteten Daten können an diesem Ort angeholt und je nach Verteilungsweg weiterverarbeitet werden.



Hinweis: An den Daten des Builder-Projekts ändert dieser Arbeitsschritt nichts, das Builder-Projekt bleibt nach dem Publizieren weiterhin gültig und kann auch verändert werden.

5.4.2 Way of Distribution

Die Freischaltung kann für zwei Verteilungswege vorgenommen werden:

- Offline
- Online

Bei der **Offline**-Verteilung wird das gesamte Scout-Projekt auf einen Client-Rechner kopiert und dort über die *Scout.exe* gestartet. Die Übertragung kann mittels geeigneter Datenträger (DVD, USB-Stick, FTP-Server etc.) erfolgen.

- *Stand Alone:*

Erstellt eine interaktive 3D VR Szene auf Basis der Unity 3D Umgebung (<https://unity3d.com/unity>). Das Scout-Projekt ist ein eigenständig lauffähiges Programm, das ohne weitere Installation auf jedem modernen Windows-PC gestartet werden kann. Neben der freien Navigation in der Szene können

beispielsweise auch Schattenanimationen, Attributabfragen und Messungen im 3D Raum, ohne weitere Vorberechnungen stattfinden.

- **Virtual Reality:**

Bereitet das Scoutprojekt zur Darstellung in VR-Brillen wie Oculus Rift und HTC Vive vor.

Die **Online** Verteilung nutzt eine Client-Server Architektur um die Scout-Daten zum Anwender zu bringen. Das Scout-Projekt wird auf einen Webserver (Apache oder IIS) gehostet und über eine Internetverbindung an den Client übertragen. Am Clientrechner werden die Daten über einen gängigen Webbrowser visualisiert. Die Software erkennt eigenständig den aktuell angezeigten Bereich und lädt genau die benötigten Daten nach (Streaming). Für die Übertragung auf mobile Endgeräte (Handy, Tablets) steht eine eigene App zur Verfügung, die über die offiziellen App-Stores (Google Play bzw. Apple App Store) bezogen werden kann.



***Hinweis:** Bei der Übertragung der Scout-Daten fallen beträchtliche Datenmengen an, die insbesondere bei Nutzung auf mobilen Endgeräten rasch zur Ausschöpfung des Datenvolumens führen. Die Nutzung innerhalb eines drahtlosen Netzwerks (WLAN) ist daher dringend angeraten.*

- **Web Browser:**

Bereitet die Scout-Daten für die Darstellung in gängigen Web Browsern (Google Chrome, Mozilla Firefox, Microsoft Internet Explorer etc.) auf. Die Anzeige im Browser erfolgt unter Nutzung von WebGL (bevorzugte Lösung) oder Java, je nachdem welche Technologie der Browser unterstützt.

- **Mobile (Cell Phone, Tablet):**

Richtet die Scout-Daten zur Nutzung in einer eigenen Scout-App her, um Scout-Projekte auf mobilen Endgeräten nutzen zu können. Die Apps stehen für alle gängigen mobilen Betriebssysteme (Android, iOS, Windows Mobile) zur Verfügung und können über die jeweiligen App-Stores bezogen werden. Die Verknüpfung des Scout-Projekts mit der App wird über den Server, der die Daten hostet, gesteuert.




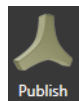
***Hinweis:** Um diese Art der Verbreitung Ihrer Scout-Projekte durchführen zu können, nehmen Sie bitte Kontakt mit UVM Systems GmbH auf. Sie erhalten dann Unterstützung bei der Einrichtung des Webserver, sowie der Einbettung Ihrer Scout Projekte.*

5.4.3 Freischaltung eines Scouts

1. Ins Builder Control Center muss ein Builder-Projekt mit berechnetem Scout (vgl. IV 5.1) eingelesen werden.



2. Am Reiter **Build** ist der Button **Distribute** zu wählen.
3. Über den Button  wird der Speicherort des Scout-Verzeichnisses angegeben.
4. Das Freischalten wird über den Button **Publish** gestartet. Je nach gewähltem Verteilungsweg und dessen Untereinstellungen werden die Scout-Daten aufbereitet. Abhängig davon ob ein einfacher Kopiervorgang oder eine aufwendige Reorganisation des Scouts notwendig ist, variiert der Zeitbedarf dieses Arbeitsschritts.



***Hinweis:** Vor dem Publizieren ist es empfehlenswert, den Scout über Config (vgl. 5.3) bereits korrekt konfiguriert zu haben. Zwar kann ein Scout auch nach dem Publizieren noch angepasst werden, jedoch bleiben die Adaptierung nur dann im Builder-Projekt und somit für erneute Publizierung erhalten, wenn der Scout sich auch noch im Builder-Projekt befindet. Wurde der Scout aber an einem anderen Ort publiziert, besteht keine weitere Verbindung zum Builder-Projekt mehr.*

5. Um lokale Änderungen mit Online Scouts zu synchronisieren ohne die Daten erneut publizieren zu müssen ist der Button **Adjust** zu klicken. Dies gilt auch für das Synchronisieren von Einstellungen, die über die **Config** Seite (vgl. IV 5.3) gesetzt wurden.



***Hinweis:** Freigeschaltete Scouts dürfen nur in einem begrenzten Umfang nachträglich verändert werden. Falls zu starke Änderungen an den Daten festgestellt werden, ist eine erneute Freischaltung des Projekts notwendig.*

6 Settings

Unter Settings sind Funktionen zum Betrieb des Builder Control Centers aufgeführt, sowie die Einstellungen für die Nutzung der Multi Core Unterstützung angesiedelt.

6.1 Multi Thread

Um die Optimierung von Builder-Daten für den Scout zu beschleunigen kann der Optimierungsprozess (vgl. IV 5.1) Arbeitsschritte parallel verarbeiten. Dieses Prinzip wird als Multi Thread Verarbeitung bezeichnet. Durch Setzen der Checkbox *Use multicore threads* aktiviert man die Parallelverarbeitung. Die Auswahlliste *Threading quality* legt nun den Grad der Parallelisierung fest

- *Low*: Ermittelt die Anzahl der CPU Kerne und erlaubt je Kern genau einen Arbeitsprozess (Thread).
- *Medium*: Je CPU Kern wird maximal die Anzahl an CPU Kernen als Threads zugelassen (z.B. bei 4 Kernen werden $4*4 = 16$ Threads erlaubt), aber nie mehr als ein Viertel der durch das Betriebssystem maximal zulässigen Threads.
- *Full*: Die Anzahl der Threads wird durch das Betriebssystem festgelegt und wird die volle Rechnerkapazität des Computers ausnutzen.



Hinweis: Multi Thread sollte nur dann genutzt werden, wenn keine weiteren Programme auf dem Computer ausgeführt werden, da der Ressourcenverbrauch durch den Builder, und die von ihm gesteuerten Subprozesse sehr umfangreich werden kann. Insbesondere bei Full konsumiert der Builder die gesamten Systemressourcen! Aus diesem Grund ist das Auslagern des Optimierungsprozesses an einen eigenständigen Arbeitsrechner angeraten. Sollte kein eigener Rechner zur Verfügung stehen sollte Multi Thread nur mit der Einstellung Low genutzt werden.

6.2 Manual



Über den Button **Manual** öffnet sich das Builder-Handbuch. Das Builder Control Center läuft im Hintergrund weiter und kann parallel zum Handbuch betrieben werden.



Hinweis: Zur Anzeige der Handbücher muss ein pdf Reader zur Verfügung stehen.

6.3 Info



Der Button **Info** öffnet ein Fenster mit allgemeinen Informationen zur CityGRID® Installation. Im Supportfall fügen Sie bitte einen Screenshot dieses Fensters an das Mail mit der Fehlerbeschreibung an.

6.4 Reset



Über den **Reset** Button werden alle Logdateien des geladenen Builder-Projekts geleert. Bitte setzen Sie alle Logs zurück, bevor ein Fehlverhalten reproduziert wird.

V. Fehlerbehandlung

Die Software CityGRID® wird von UVM Systems GmbH mit dem Streben nach Fehlerfreiheit entwickelt, getestet und gewartet. Dennoch können wir nicht ausschließen, dass während der Bearbeitung Fehler auftreten.

Folgende Fehler können auftreten:

- Absturz von Autodesk MAX ohne vorheriger Fehlermeldung: Die Ursache für derart fatale Fehler liegt meist bei MAX selbst.
- Fehlermeldungen von CityGRID®: Die Ursache liegt oft bei fehlerhaften Daten etc. Es kann normal weitergearbeitet werden.
- Script-Fehler: Die Ursache liegt oft an einer Fehlbedienung. Meist kann normal weitergearbeitet werden. Falls Folgeprobleme auftreten: Plugin CityGRID® schließen und neu starten.

Nach dem Auftreten von Abstürzen kann es sein, dass temporäre Daten auf der Festplatte liegen bleiben. Entfernen Sie insbesondere nach Abstürzen im TMP-Verzeichnis ihres Benutzernamens, große Dateien. Sie finden dorthin z.B. durch Eingabe von „cd %TMP%“ in einem Kommandofenster.

Falls Sie einen Fehlerbericht senden wollen, bitte folgende Informationen/Daten mitschicken:

1. Beschreibung der Aktionen, die zum Fehler im 3D Studio Builder geführt haben
2. Protokolldatei: Sie finden die Protokolldatei in Ihrem Profil unter \AppData\Roaming\CityGRID\CityGRID.log.
3. Jedes Builder-Projekt speichert sein Protokoll namens SuGu.log im Builder-Projektordner.
4. Das Scout-Log, auch SuGu.log genannt, befindet sich im Builder-Projektordner unter Scout / Content / Data / SuGu



Hinweis: Diese Datei bitte sofort nach dem Absturz sichern, da ein neuerlicher Programmstart zum Überschreiben dieser Datei führen kann.



Tipp: Im 3D Studio Builder Plugin kann das Verzeichnis der Protokolldatei direkt über das Menü „CGBuilder – Protokollierung“ geöffnet werden.

Die Fehlerberichte senden Sie bitte an support@uvmsystems.com

VI. Kontakt



UVM
SYSTEMS

UVM Systems GmbH

www.citygrid.at

www.uvmsystems.com