

IDM-3D. Connection

Ekkehard Beier (Editor)

Version

Spezifikation: IDM-3D.Connection 1.0

Dokumentenversion: Finale Version

Copyright: Das Urheberrecht fuer das dieser Spezifikation zugrundeliegende Anfuuegepunktmodell liegt und verbleibt bei der intelligentgraphics AG. Daraus ergeben sich keine Einschränkungen fuer das Nutzungsrecht dieser Spezifikation durch Dritte aus Sicht der intelligentgraphics AG.

Spezifikation

IDM-3D.Connection

IDM-3D.Connection ist ein Bestandteil der IDM-3D-Spezifikation und beschreibt die Definition von Anfuuegepunkten, nachfolgend auch 3D-Anfuuegepunkte, welche zu Verbindung von Produkten im Sinn von IDM Living verwendet werden. Typische Beispiele von Produktverbindungen sind:

- Symmetrische Links-Rechts-Verbindungen, z.B. eine Polsterzusammenstellung
- Asymmetrische Kindverbindung, z.B. eine Kopfstuetze (als separates Produkt) an einem anderen Produkt

Anfuuegepunkte sowie die Relationen zwischen Anfuuegepunktpaaren sind prinzipiell dynamisch und werden in ihren Parametern bzw. Gueltigkeit durch bestimmte Zustaeude beeinflusst:

- Eine Intraproduktregel beeinflusst die Parameter eines Anfuuegepunkts: Je nach Breite der gewaehlten linken Armlehne ergibt sich die Breitekoordinate des rechten Anschlusspunktes. Je nach Wahl des Merkmals ‚mit/ohne Kopfstuetzenhalterung‘ ist der Anfuuegepunkt einer Kopfstuetze aktiv oder inaktiv.

- Eine explizite Interproduktregel beruecksichtigt mindestens zwei Produkte um deren Anfuegepunkte zu parametrisieren: Couch-Elemente koennen nur verkettet werden, wenn ihre Sitzhoehe gleich ist.
- Eine generische Interproduktregel ist unabhaengig von konkreten Produkten definiert: ein Produkt kann trotz gueltiger Paarigkeit nicht angeplant werden, weil sich dann eine Kollision mit einem anderen Produkt ergeben wuerde.

Somit ergibt sich ein komplexer Gesamtalgorithmus, welcher nicht vollumfassend beschrieben werden kann, weil es einerseits unterschiedliche Szenarien geben kann (rein kaufmaennische Auftragserfassung, 2D-Planung auf Basis von abstrahierten Symbolen, 3D-Planung mit realistischer Kollisionserkennung). Andererseits ist ein solcher Gesamtalgorithmus auch Teil einer spezifischen Software-Loesung und damit Gegenstand des Wettbewerbs.

Insofern stellen IDM Living und IDM-3D.Connection lediglich Bausteine, wenn auch sehr wichtige, fuer einen letztlich spezifischen Gesamtalgorithmus dar:

- IDM Living beschreibt die prinzipielle Paarigkeit von Anfuegepunkten.
- IDM Living definiert in 2D-Draufsichtssymbolen Anschlussvektoren mit einer festgelegten Semantik und statischen oder dynamischen Koordinaten.
- IDM Living beschreibt die nominalen Abmessungen, welche durch Anwendung auf dynamische 2D-Draufsichtssymbole zusammen mit den in diesen definierten Anschlussvektoren konkrete Koordinaten ergeben.
- IDM-3D.Connection beschreibt parametrische Anfuegepunkte - sowie die Paarigkeiten zwischen diesen ueber generische Schluesselpaare oder Tags.

Somit ergibt sich eine Ueberlappung von Daten einerseits fuer den Bereich der Paarigkeiten, andererseits fuer den Bereich der Koordinaten. Im Bereich der Paarigkeiten soll IDM-Living das fuehrende System sein, welches eventuell durch IDM-3D.Connection verfeinert wird. Ebenfalls ist es denkbar, IDM-3D.Connection als Schnellvariante waehrend interaktiver Bewegungen zu verwenden und erst beim Abschluss der Bewegung eine entsprechend aufwaendige, in der Regel serverbasierte Auswertung durchzufuehren. Fuer die Ueberlappung im Bereich der Koordinaten wird weiter unten eine Festlegung getroffen.

Ein IDM-3D.Connection-Datensatz ist vorzugsweise eine vollumfaengliche Zusammenstellung der Anfuegepunkte eines Projekts, wobei die Zuordnung eines Projekts zu einem Hersteller, einer Produktserie, einer Teilproduktserie, mehreren Produktserien [auf kaufmaennischer Ebene] nicht definiert ist. So kann es beispielsweise sinnvoll sein, Anfuegepunkte bezugnehmend auf einen (1) IDM Living-Datensatz zusammenzufassen. Anfuegepunkte werden einem (1) Produkt zugeordnet.

Es besteht eine funktionale Ueberlappung zwischen IDM-3D.Connection und IDM Living (incl. der SVGs), nachfolgend 2D-Anfuegepunkte. Diesbeueglich wird definiert:

- Im 2D-Kontext haben die 2D-Anfuegepunkte Vorrang. Fehlen diese und sind 3D-Anfuegepunkte vorhanden, werden letztere verwendet.
- Im 3D-Kontext haben die 3D-Anfuegepunkte Vorrang. Fehlen diese und sind 2D-Anfuegepunkte vorhanden, werden letztere verwendet.

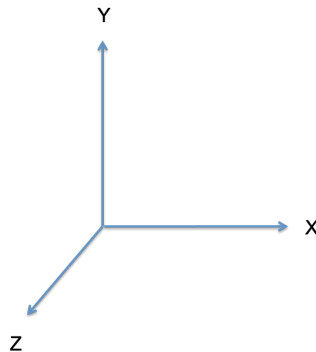
Bezug

Anfuegepunkte werden relativ zum Produktursprung definiert. Dieser Ursprung ist im Allgemeinen: links, unten und hinten – wobei Ueberstaende durch Funktionen/Animationen ignoriert werden¹.

Insbesondere fuer Anbauteile wie Kopf- oder Fusstuetzen empfehlen sich dagegen andere Bezuege – in der Regel auf der Symmetrieachse.

Hoehenplazierte Anbauteile sollten ihrerseits hoeehenneutral modelliert werden und erst ueber ein entsprechendes Anfuegepunktpaar nach oben verschoben werden.

Das Bezugssystem hat die Maszeinheit 1 Meter. Das heisst: 0.5 korrespondiert zu 0,5 Meter. Das zugrundegelegte Koordinatensystem ist wie folgt definiert:



- x-Achse: nach rechts [aufsteigende Werte]
- y-Achse: nach oben
- z-Achse: nach vorn

Die Interpretation der Rotation ergibt sich entsprechend der Rechte-Hand-Regel.

¹ Diese Festlegung ist aus Sicht der 3D-Modellierung, insbesondere fuer die Massendatenanlage vorteilhaft. Dies widerspricht nicht der Verwendung eines x-z-Mittelpunkts fuer das interaktive Drehen von Produkten, da solche Mittelpunkte auch dynamisch ermittelt werden koennen bzw. muessen (Multiproduktauswahl).

Parametrik

Die Definition von Anfruegepunkten erfolgt prinzipiell parametrisch. Das heisst, erst der tatsaechliche Konfigurationszustand praegt letztlich den Anfruegepunkt aus, was auch seine Aktivierung mit einschliesst. Demzufolge sind bestimmte Merkmale eines Anfruegepunkts entweder fixe Werte des angegebenen Typs (Boolean, Zahl) oder ein Ausdruck, dessen Evaluierung einen Wert des angegebenen Typs ergibt.

Ausdruecke werden als Strings angegeben, die in einen Block eingebunden sind, z.B. "{ 0.4 * 0.5 }"². Der Inhalt des Ausdrucks wird als JavaScript interpretiert (ECMA 5.1). Der Evaluierungskontext muss eine Funktion \$s bereitstellen, welche den Zugriff auf den Konfigurationszustand erlaubt.

Die Funktion \$s sei wie folgt definiert:

`$s(key: string, default: string): string`

Dabei ist *key* der Identifier des relevanten Merkmals und *default*, der Wert, der geliefert werden soll, falls fuer dieses Merkmal kein Wert im Konfigurationszustand existiert.

Die Laufzeitumgebung muss den Konfigurationszustand als flache Liste bereitstellen und hierbei sowohl Merkmalsidentifier wie auch -werte als Strings uebergeben. Numerische Werte muessen in der wissenschaftlichen Notation konvertiert werden, Boolesche Werte als True oder False.

Weiterhin koennen nutzerdefinierte Funktionen deklariert werden, um Ausdruecke kompakter zu gestalten. Die Funktionen duerfen keinen Seiteneffekt verursachen, z.B. durch Modifikation globaler Variablen oder Funktionen, wie \$s.

Anfruegepunktparameter

Parameter, die parametrisch sind, sind durch ein {} gekennzeichnet. Zahlen muessen in wissenschaftlicher Notation angegeben werden.

Key: string

Der Parameter definiert den primaeren Schluessel des Anfruegepunkts im Kontext eines Produkts. Er wird auch verwendet um Anfruegepunktpaarigkeit zu definieren/festzustellen.

² Das erste und letzte Zeichen sind also fuer den Block reserviert und kennzeichnen damit auch die Notwendigkeit der Evaluierung des Ausdrucks – im Gegensatz zu einem direkten Parsen des Zieltyps.

Der Parameter ist zwingend erforderlich und ist auf 10 ASCII-Zeichen (A ... z, 0 ... 9, _, -) limitiert.

Foreign: string

Der Parameter definiert eine Maske fuer Anfuuegepunkte, die an diesem Anfuuegepunkt platziert werden koennen. Das Zeichen * kann hierbei verwendet werden, um mehrere Keys zuzulassen. Die Maske Links_* erlaubt z.B. die Keys Links_1 und Links_2.

Der Parameter ist zwingend erforderlich und ist auf 10 ASCII-Zeichen (A ... z, 0 ... 9, _, -) limitiert. Ausserdem kann das Zeichen * verwendet werden.

Tags: string[]

Der optionale Parameter definiert eine Liste von Tags um eine zusaetzliche feingranulare Anfuuegepunktpaarigkeit auszudruecken. Ein Tag ist hierbei eine freie Zeichenkette mit den Zeichen (A ... z, 0 ... 9, _, -). Zwei Tags sind identisch, wenn ihre Normalisierung auf Kleinbuchstaben identisch ist.

Der Parameter wird entweder als string-serialisiertes JSON-Array von strings angegeben oder als parametrischer Ausdruck, dessen Evaluierung ein string-Array ergeben muss.

Type: string

Der Parameter definiert den Typ des Anfuuegepunkts. Er wird auch verwendet um Anfuuegepunktpaarigkeit zu definieren/festzustellen.

Der Parameter ist zwingend erforderlich und muss einen der folgenden Werte haben:

- Child – Anfuuegepunkt, an welchem ein Produkt als Anbauelement angeplant werden kann.
- Bottom – Anfuuegepunkt, an welchem ein Produkt als vertikaler Nachbar angeplant werden kann. Der Anfuuegepunkt sollte unten sein, wenn man auf das Produkt schaut.
- Left – Anfuuegepunkt, an welchem ein Produkt als horizontaler Nachbar angeplant werden kann. Der Anfuuegepunkt sollte links sein, wenn man von vorn auf das Produkt schaut.
- Origin – Ursprung des Produkts in Bezug auf ein Anfuuegepunktpaar.

- Right – Anfügepunkt, an welchem ein Produkt als Nachbar angeplant werden kann. Der Anfügepunkt sollte rechts sein, wenn man von vorn auf das Produkt schaut.
- Top – Anfügepunkt, an welchem ein Produkt als vertikaler Nachbar angeplant werden kann. Der Anfügepunkt sollte oben sein, wenn man auf das Produkt schaut.

Reserviert: Front, Back, Alignment, Minimum, Maximum

Die Anfügepunktpaarigkeiten auf Typebene sind dann:

- Origin + Child – Eine asymmetrische Beziehung, in der sich Manipulationen des Origin-Produkts eher auf das Child-Produkt auswirken als umgekehrt.
- Left + Right – Eine symmetrische Beziehung, in der sich Manipulationen des Left-Produkts ebenso auf das Right-Produkt auswirken, wie auch umgekehrt.
- Top + Bottom – Eine symmetrische Beziehung, in der sich Manipulationen des Top-Produkts ebenso auf das Bottom-Produkt auswirken, wie auch umgekehrt.

Position: {}

Der Parameter definiert die Position des Anfügepunkts relativ zum Ursprung des Produkts, welcher in der Regel aber nicht zwangsläufig links – hinten – unten ist.

Position.X: number {}

Der Parameter definiert die X-Koordinate des Anfügepunkts in Meter. Der Default-Wert ist 0.

Position.Y: number {}

Der Parameter definiert die Y-Koordinate des Anfügepunkts in Meter. Der Default-Wert ist 0.

Position.Z: number {}

Der Parameter definiert die Z-Koordinate des Anfügepunkts in Meter. Der Default-Wert ist 0.

Rotation: {}

Der Parameter definiert die Position des Anfügepunkts relativ zum Ursprung des Produkts, welcher in der Regel aber nicht zwangsläufig links – hinten – unten ist.

Rotation.X: number {}

Der Parameter definiert die exklusive Verdrehung um die X-Achse in Grad. Der Default-Wert ist 0.

Rotation.Y: number {}

Der Parameter definiert die exklusive Verdrehung um die Y-Achse in Grad. Der Default-Wert ist 0.

Rotation.Z: number {}

Der Parameter definiert die exklusive Verdrehung um die Z-Achse in Grad. Der Default-Wert ist 0.

Enabled: boolean {}

Der Parameter definiert, ob der Anfuegepunkt aktiv oder inaktiv ist.

Representation: string

Der Parameter legt eine visuelle Repraesentation fuer den Anfuegepunkt fest.

Visuelle Repraesentationen koennen gegenfalls farblich parametrisiert werden. Sie muessen von der Kollisionsermittlung ausgeschlossen werden.

Definiert sind die folgende Werte:

- Standard – Standard- bzw. Default-Repraesentation
- Ghost – Es sollte an diesem Anfuegepunkt eine sogenannte Geisterdarstellung eines dort anzufuegenden Produkts angezeigt werden.
- Ghost2 – Es sollte an diesem Anfuegepunkt eine sogenannte Geisterdarstellung eines dort anzufuegenden Produkts angezeigt werden, allerdings soll eine leichtgewichtige Version verwendet werden³.
- HotSpot – Es handelt sich um einen Pseudoanfuegepunkt, der fuer spezielle Zwecke verwendet werden soll.

Anfuegepunktpaarigkeit

Insgesamt ergibt sich eine Anfuegepunktpaarigkeit wenn:

³ Im Bereich von Regalplanungen kann z.B. Ghost fuer ein Makroraster und Ghost2 fuer ein Mikroraster verwendet werden.

- Key und Foreign passen, ggf. unter Einbeziehung der Wildcard-Semantik,
- es mindestens einen identischen Tag auf beiden Seiten gibt, insofern mindestens ein Anfügepunkt Tags definiert,
- die Typen passen, wie oben beschrieben.
- beide Anfügepunkte aktiviert sind,
- die ins globale Koordinatensystem transformierten Anfügepunkte
 - einen Positionsoffset < 0.2 [m]
 - einen Rotationsoffset < 45 Gradhaben (wobei es sich hier um Richtwerte handelt, die anders oder auch adaptiv gehandhabt werden können)

Referenzen

[ECMA5.1] ECMAScript Language Specification, <https://262.ecma-international.org/5.1/>

[SCHEMA] <https://archive.intelligentgraphics.biz/schemas/idm-3d/connection.json>

Historie

IDM-3D.Connection 1.0 – 06.10.2023

Kontakt

Daten Competence Center e. V.

Goebenstraße 4-10

32052 Herford

Info: www.dcc-moebel.org

Dr.-Ing. Olaf Plümer

E-Mail: pluemer@dcc-moebel.org

Tel.: +49 52 21 / 12 65-37

PD Dr. Ing. habil Ekkehard Beier

E-Mail: eb@intelligentgraphics.ag

Tel.: +49 36 77 / 66 73 520