Marco Tarini - Computer Graphics 2025/2026
Università degli Studi di Milano

Mesh poligonali: alcune caratteristiche

66

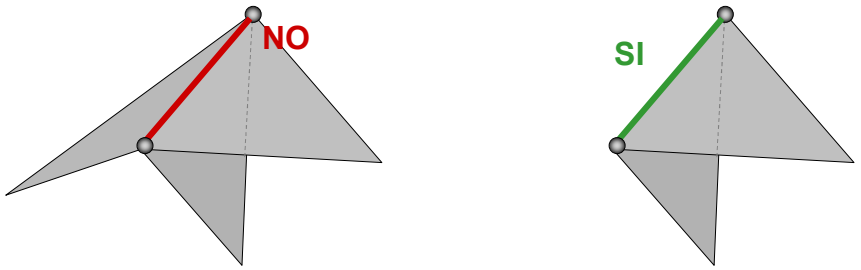
Mesh two-manifold e non

- ✓ una mesh è detta two-manifold (una "varietà due") quando *rappresenta in effetti una superficie*
  - ⇒ molti algoritmi di geometry processing necessitano che questo sia il caso!
- ✓ Non tutte le mesh (= insiemi di poligoni che condividono dei vertici e degli edge) lo sono!
  - ⇒ le facce di una mesh rappresentano sempre (pezzi di) superficie – tutto ok
  - ⇒ su edge e vertici le cose possono andare storte
  - ⇒ vediamo quali condizioni devono verificare gli edge e i vertici affinché la mesh rappresenti una superficie bidimensionale?

67

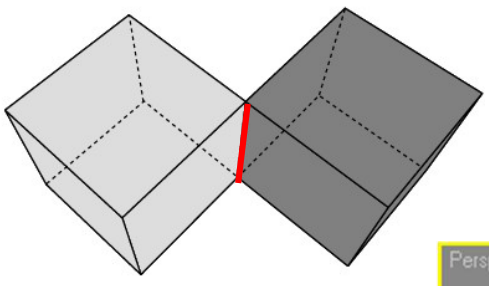
Edge two manifold

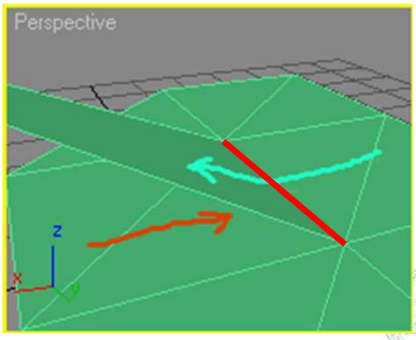
- ✓ un edge two-manifold se è condiviso da al più due facce



68

Esempi di edge non 2-manifolds



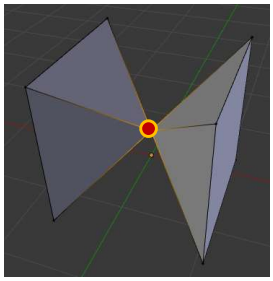


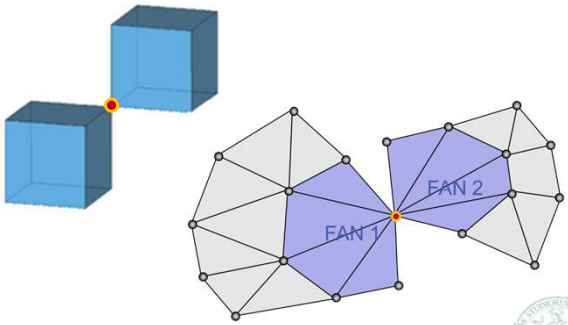
69

Vertice two-manifold

- ✓ Due facce sono dette *adiacenti* se condividono un edge
- ✓ L'insieme di facce che condividono un vertice, tutte adiacenti a coppie = un fan (letteralmente: un ventaglio)
- ✓ un vertice è two manifold se ha un solo fan di facce

Esempi di vertici non 2-Manifold:

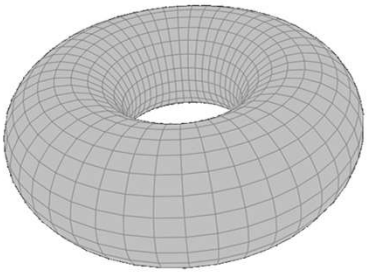




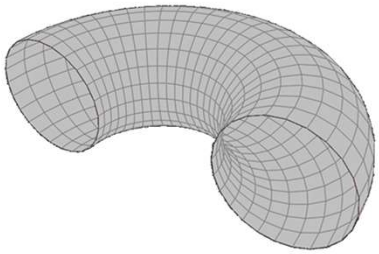
72

Mesh chiuse e aperte

- ✓ un edge condiviso da 1 faccia è «aperto», o di bordo;
- ✓ un edge condiviso da 2 faccie è «interno»;
- ✓ se una mesh non ha edge di bordo, è chiusa
- ✓ altrimenti, è aperta
  - ⇒ la distinzione ha senso solo se la mesh è two-manifold



chiusa

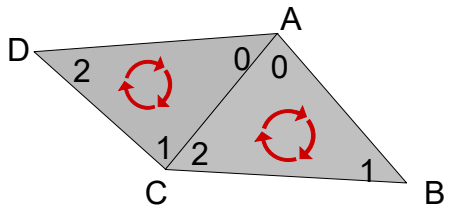


aperta

75

Orientamento delle facce

- ✓ Una mesh two manifold può essere «ben orientata» oppure no
- ✓ Ben orientata = l'ordinamento dei tre vertici dentro ogni faccia è consistente (sempre senso orario oppure sempre senzo antiorario)

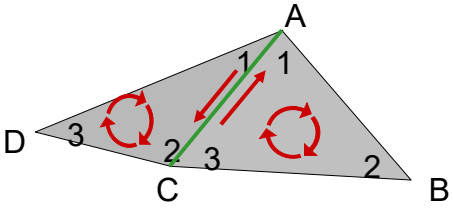


t_a = {A,C,D}
t_b = {A,B,C}

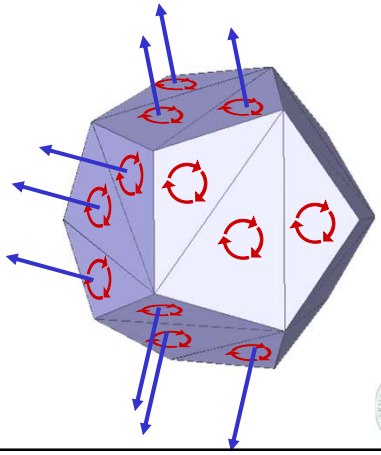
76

Mesh

- ✓ Ben orientata, non ben orientata
  - ⇒orientamento di ogni faccia coerente?
  - ⇒ben orientata → normali coerenti



senso opposto, edge coerente



77

Osservazione

```
OFF
12 10 0
0.0 0.0 0.0
3.0 0.0 0.0
3.0 1.0 0.0
1.0 1.0 0.0
1.0 5.0 0.0
0.0 5.0 0.0
0.0 0.0 1.0
3.0 0.0 1.0
3.0 1.0 1.0
1.0 1.0 1.0
1.0 5.0 1.0
0.0 5.0 1.0
4 3 2 1 0
4 5 4 3 0
4 6 7 8 9
4 6 9 10 11
4 0 1 7 6
4 1 2 8 7
4 2 3 9 8
4 3 4 10 9
4 4 5 11 10
4 5 0 6 11
```

geometria

connettività

Le caratteristiche di essere:
two-manifold, chiusa, aperta, ben orietnata, bel orientabile…
dipendono esclusivamente dalla connettività della mesh
(non la sua geometria). Per es:

Esercizio:
Puoi dire, guardando il testo del file ←qui accanto, se si tratti una mesh:

- quad-dominant, pure-quad, o tri?
- two-manifold o no?
- chiusa o aperta?
- ben orientata o no?
- low-poly o hi-res?

78

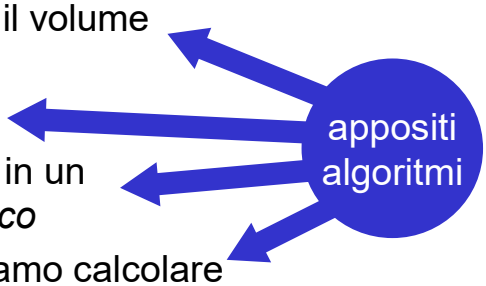
Note sull'Esercizio

- ✓ Ogni coppia di indici consecutivi in ogni faccia individua un edge
  - ⇒ compreso: dall'ultimo vertice della faccia al primo. L'array è ciclico!
- ✓ L'edge è two-manifold se: compare in max due facce.
- ✓ L'edge è chiuso se: compare in due facce
- ✓ L'edge è aperto se: compare in una faccia sola
- ✓ Due facce che condividono un edge sono orientate consistentemente se:
  l'edge appare, nei due casi, in versi opposti
  (es: 1 => 5 nella prima faccia, e 5 => 1 nella seconda)
- ✓ Edge di bordo (o «aperto») se: appare solo in una faccia
- ✓ Mesh two-manifold se: tutti gli edge sono two-manifold
- ✓ Mesh chiusa se: non ci sono edge aperti
- ✓ Mesh ben orientata se: tutti le coppie di facce adiacenti sono orientate consistentemente

80

Osservazione

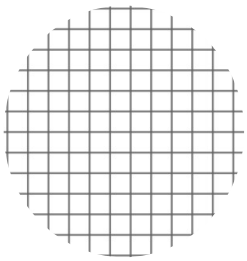
- ✓ Una mesh two-manifold, ben-orientata e chiusa separa uno spazio *interno* da uno *esterno*
- ✓ Quindi:
  - ⇒ Rappresenta un oggetto solido!
  - ⇒ Possiamo calcolarne il volume (oltre che l'area)
  - ⇒ …e il suo baricentro
  - ⇒ Possiamo convertirla in un modello 3D *volumetrico*
  - ⇒ Dato un punto, possiamo calcolare se è esterno o interno
  - ⇒ Possiamo stamparla in 3D!



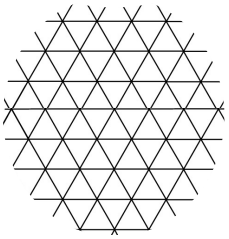
81

Regolarità di una mesh: concetto

- ✓ Osservazione: un piano può essere tassellato in modo **regolare** da quadrati ==>
- ✓ Tanto più la *connettività* di una **quad-mesh** si avvicina a questo caso, tanto più la consideriamo "**regolare**"



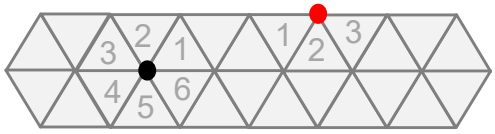
- ✓ Osservazione: un piano può essere tassellato in modo **regolare** da triangoli equilateri ==>
- ✓ Tanto più la *connettività* di una **tri-mesh** si avvicina a questo caso, tanto più la consideriamo "**regolare**"

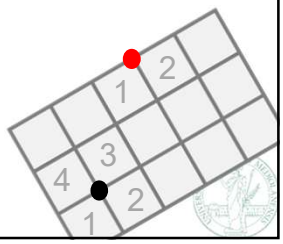


84

Regolarità di una mesh definizioni

- ✓ Vertice interno = vertice che non compare su un edge aperto
  - ⇒ Altrimenti: il vertice è di bordo
- ✓ «Valenza» di un vertice
  - ⇒ numero di facce adiacenti ad quel vertice
  - ⇒ a volte: numero di edge adiacenti ad quel vertice
  - ⇒ per tutti i vertici interni: le due definizioni sono equivalenti
- ✓ «Vertice regolare» interno = un vertice di valenza…
  - ⇒ …4, in una quad-mesh
  - ⇒ …6, in una tri-mesh
- ✓ «Vertice regolare» di bordo = un vertice di valenza…
  - ⇒ …2, in una quad-mesh (2 facce, e 3 edge)
  - ⇒ …3, in una tri mesh (3 facce, e 4 edge)





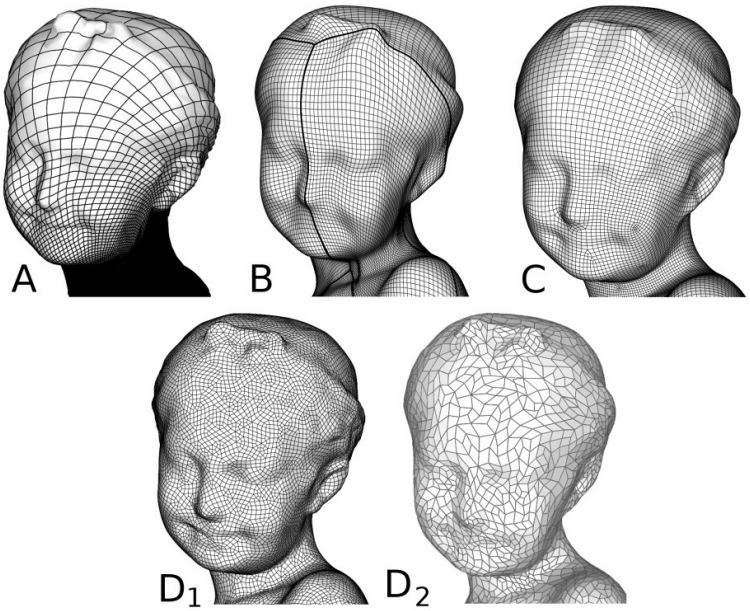
87

Regolarità di una mesh definizioni

- ✓ Quanti vertici in una mesh (tri o quad) sono regolari?
  - ⇒ Tutti => la mesh è (*perfettamente)* regolare
    (è detta anche «structured» mesh)
  - ⇒ Quasi tutti - per es il 99% => la mesh è «semi-regolare»
  - ⇒ Pochi (per es, solo 2/3 o la metà) = è una mesh irregolare
  - ⇒ Nota: la regolarità di una mesh è una questione di grado
- ✓ Quad-mesh fortemente regolare = in pratica, è un grigliato

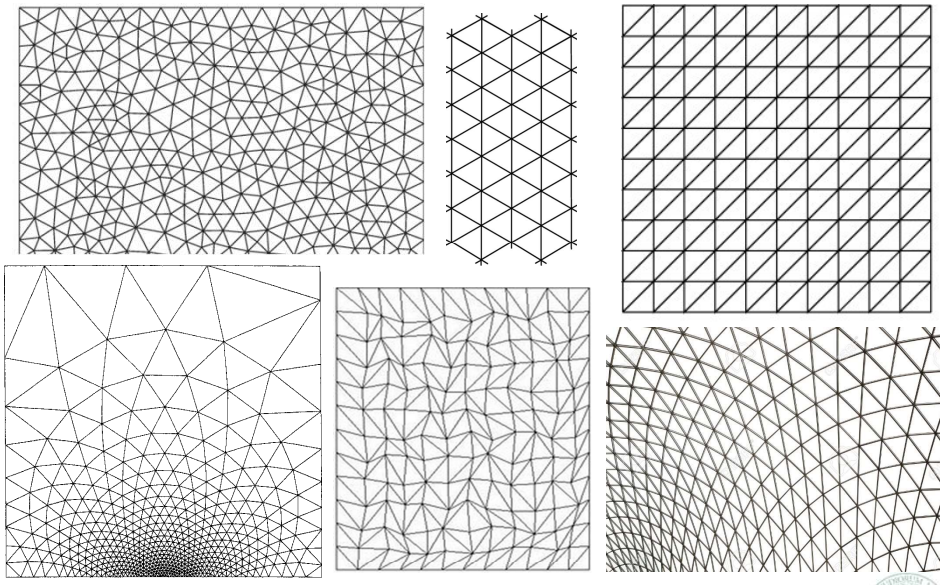
88

Mesh: la regolarità è questione di grado



89

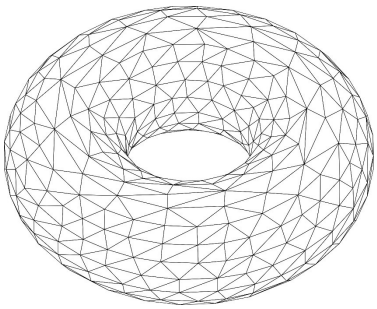
Regular VS irregular Triangular mesh



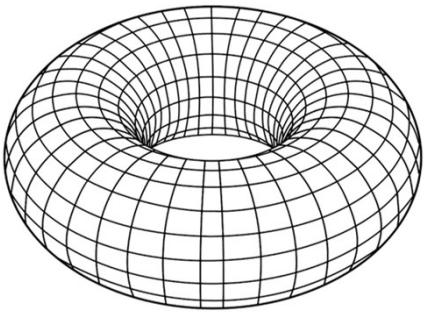
Esercizio: trova i vertici irregolari. Quali mesh sono (semi-) regolari?
Ripeti con altre mesh che vedi in questi lucidi

91

Regolare VS irregolare



una tri-mesh
irregolare

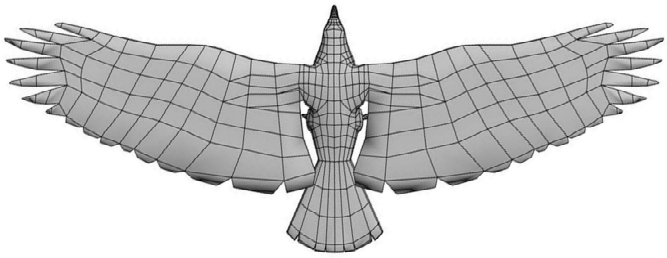


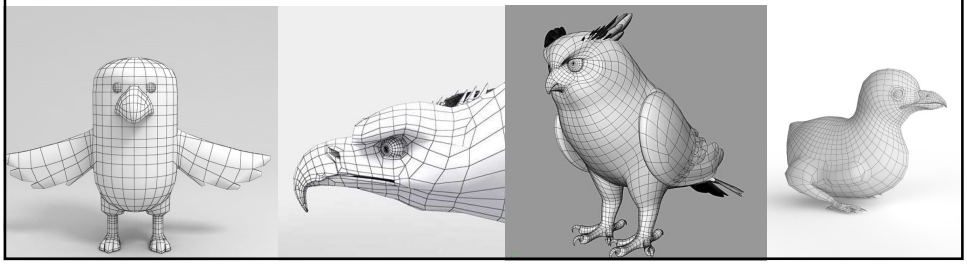
una quad-mesh
regolare

93

Regolarità delle mesh

✓ Le mesh modellate *manualmente* da modellatori (nel CAD, nel direct low-poly editing [vedi dopo]) tendono ad essere **quad-mesh regolari**

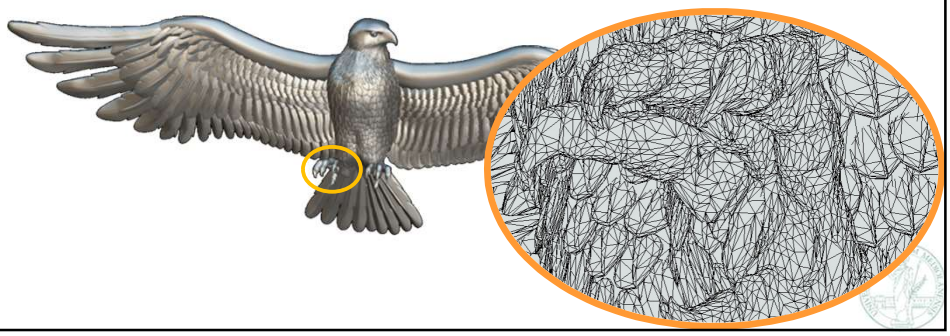




94

Regolarità delle mesh sul campo

- ✓ Sono di solito **tri-mesh irregolari** le mesh
  - ⇒ **acquisite** 3D da modelli reali (per es, scanning o fotogrammetria)
  - ⇒ prodotte o ri-processate attraverso tecniche di **geometry processing** per esempio tutti che vedremo: semplificazione automatica, front advancing methods, triangolazioni di Delaunay, vertex clustering… (eccetto ovviamente per le tecniche che si prefiggono questo obiettivo, come il semi-regular remeshing)
  - ⇒ Prodotte da artisti digitali con tecniche di **digital sculpting** (vedi dopo)



95

Molti vantaggi (e alcuni svantaggi) della regolarità

- ✓ Una mesh più regolare tende ad essere
  - ⇒ Più agevole da modificare / o ri-editare manualmente (ad esempio, consente la selezione di «stream di edge»)
  - ⇒ Più adatta ad essere animata
  - ⇒ Più efficiente da comprimere per risparmiare memoria (o da mandare in streaming in rete)
  - ⇒ Maggiormente adatta ad applicazioni di Machine Learning
  - ⇒ Spesso maggiormente accurata geometricamente, a parità di risoluzione
  - ⇒ Meno soggetta a presentare artefatti di rendering
  - ⇒ Tuttavia, la risoluzione di una mesh regolare tende ad avere una risoluzione meno adattiva: l'adattività «si paga» in termini di regolarità
- ✓ In generale, la regolarità mesh è una caratteristica desiderabile
  - ⇒ ma spesso difficile da ottenere in modo automatico
  - ⇒ Nota: per l'efficienza di rendering, non fa differenza
  - ⇒ Domanda: cosa si ottiene effettuando un diagonal-split a tutte le facce di una quad-mesh **regolare**?

96

Marco Tarini - Computer Graphics 2025/2026
Università degli Studi di Milano

3D Models - Polygonal meshes
Note sulla loro generazione

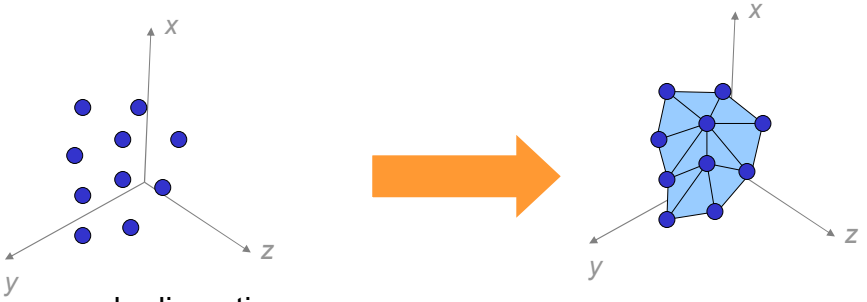
102

Generazione di mesh poligonali

- ✓ Occupiamoci ora dei metodi esistenti per *generare* **mesh poligonali**
- ✓ Una caso comune è la generazione di **tri-mesh** a partire da **nuvole di punti**
  - ⇒ Cioè la **surface reconstruction** da una point cloud
  - ⇒ Perché si tratta di ricostruire la superficie (a partire da un suo campionamento)
  - ⇒ Detto anche «meshing», in generale il processo di costruire una mesh
  - ⇒ Un task di **geometry processing**, naturalmente
  - ⇒ Ad esempio, la **fotogrammetria** genera **nuvole di punti**, che tipicamente vanno convertite in mesh per il loro utilizzo
- ✓ Un approccio possibile consiste nel considerare i punti della nuvola come la **geometria** della mesh, e creare la **connettività** identificando dei triangoli (triple di vertici) che li connettono
  - ⇒ (come sappiamo geometria = insieme di vertici)
  - ⇒ Alcuni punti della nuvola possono essere ignorati e scartati, quindi sono considerati «outliers»

103

Da Nuvola di punti a Triangle Mesh

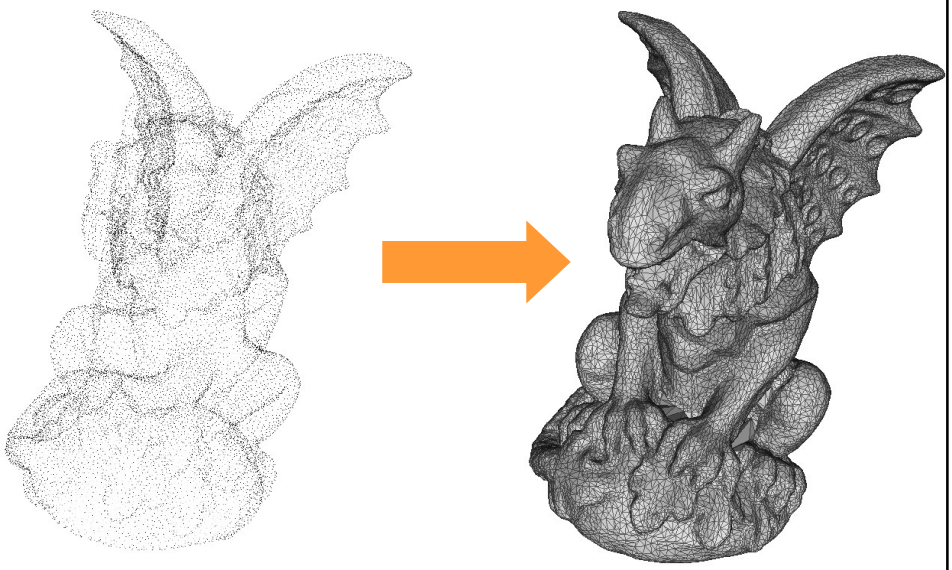


nuvola di punti

triangoli

104

Da Nuvola di punti a Triangle Mesh

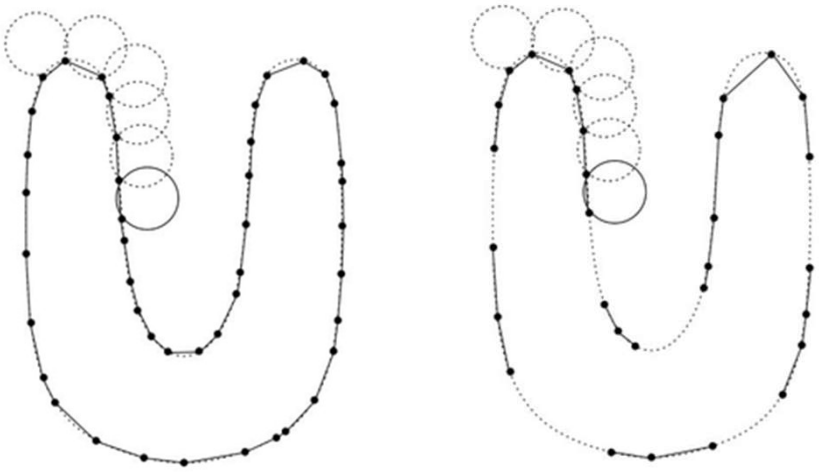


105

Da Nuvola di punti a Triangle Mesh

✓ Una classe di algoritmi: Front Advancing:

⇒Come per es «ball pivoting»



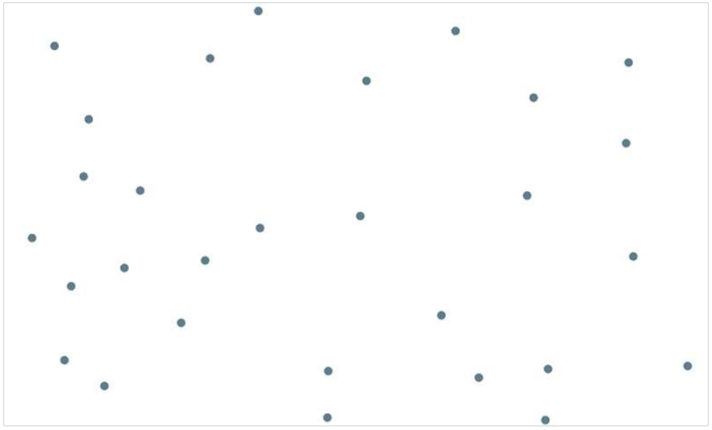
[Bernardini, Mittleman, Rushmeier, Silva TVCG 99]

106

Da Nuvola di punti a Triangle Mesh… in 2D

✓ In 2D: il problema ha un'ottima soluzione:
la triangolazione di Delaunay

⇒Ben nota dagli anni '30 (in geom. computazionale)



Vertici (sul piano x,y)

111

Da Nuvola di punti a Triangle Mesh… in 2D

- ✓ In 2D: il problema ha un'ottima soluzione: la triangolazione di Delaunay
  - ⇒ Ben nota dagli anni '30 (in geom. computazionale)

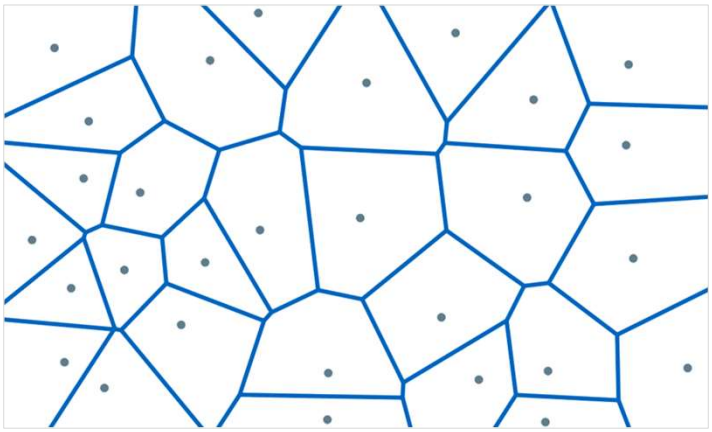
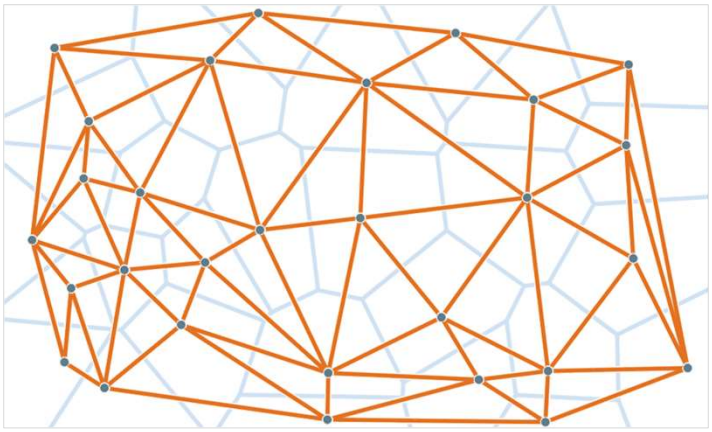


Diagramma di Voronoi

112

Da Nuvola di punti a Triangle Mesh… in 2D

- ✓ In 2D: il problema ha un'ottima soluzione: la triangolazione di Delaunay
  - ⇒ Ben nota dagli anni '30 (in geom. computazionale)



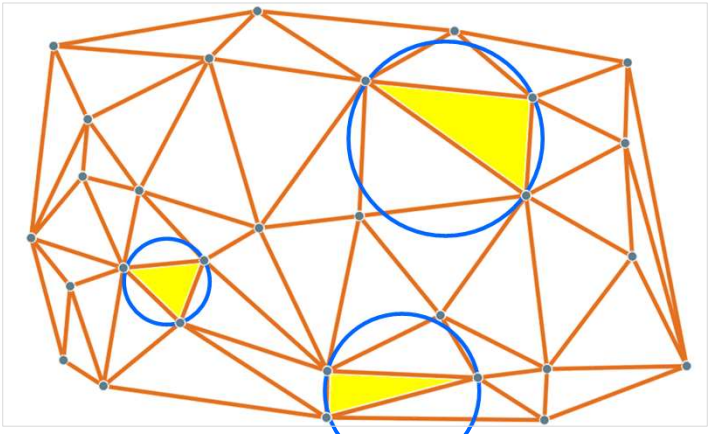
Triangolazione di Delaunay

113

Da Nuvola di punti a Triangle Mesh… in 2D

✓ E', una «buona» triangolazione

⇒Una buona proprietà è garantita:



Triangolazione di Delaunay

114

Triangolazione di Delaunay: note

✓ Diagramma di Voronoi (in 2D)

⇒Una partizione in "regioni" indotta da n punti (detti "seed") $\mathbf{s}_0, \mathbf{s}_1, \mathbf{s}_2, \ldots$

⇒Ogni seed produce una regione

⇒Regione di un seed $\mathbf{s}_i$ = insieme di punti $\mathbf{p}$ del piano che sono più vicini a $\mathbf{s}_i$ che ad ogni altro seed

✓ Triangolazione di Delaunay

⇒Il "duale" di un diagramma di Voronoi cioè:

⇒La connettività ottenuta connettendo con un edge ogni coppia di seed di regioni *confinanti fra loro*

⇒E' una triangolazione con ottime proprietà, come ad esempio: *il circocentro che inscrive ogni triangolo non include nessun altro vertice*

116

Da Nuvola di punti a Triangle Mesh passando per…

		ELEMENTI DISCRETI			CONTINUI
		regolari «a griglia»	semi-regolari o irregolari: elementi simpliciali	semi-regolari o irregolari: elementi non simpliciali	
SUPERFICIALI	2-manifold «rappresenta una vera superficie»	Height Field Range Scan (Geometry Images)	Triangle Mesh	Polygonal Mesh Quad-Mesh Quad dominant Mesh	Subdivision surface Parametric Surface (es. B-splines)
	non-manifold «non rappresenta una sup»	Set di Range Point Cloud			
VOLUMETRICI	(3-manifold)	Voxels Solid Textures	Tetra Mesh	Hexa Mesh	Implicit model (es. CSG)

118

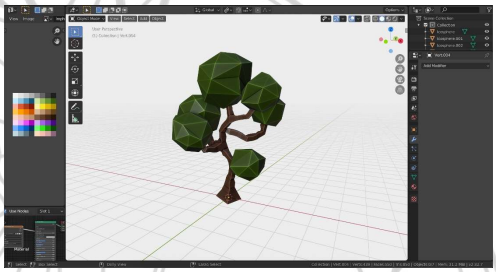
Da Nuvola di punti a Triangle Mesh

- ✓ Sommario: alcuni metodi aggiungono la connettività
  - ⇒ la nuvola di punti costituisce già la geometria della mesh (anche se è possibile scartarne alcuni)
  - ⇒ Front Advancing, come Ball Pivoting
  - ⇒ Delaunay triangulation (da Voronoi diagram) (è definita in 2D, ma possiamo estendere a 3D se i punti vengono prima proiettati su un piano)
  - ⇒ Un problema comune è che i vertici della mesh mantengono il rumore e le inesattezze della nuvola di punti
- ✓ Anticipazione: altri metodi ricampionano i vertici
  - ⇒ Uso di rappresentazioni volumetriche intermedie
  - ⇒ Come "Poisson Reconstruction"
  - ⇒ Che vedremo nella lez. su rappresentazioni volumetriche
  - ⇒ In pratica, questi metodi consentono di attenuare fortemente i problemi dovuti al noise e agli outliers della nuvola di punti

119

Marco Tarini - Computer Graphics 2025/2026
Università degli Studi di Milano

Modelli poligonali: modellazione manuale (basi)



120

Sommario: come vengono generate le mesh

- ✓ Cattura dalla realtà
  - ⇒ Cioè 3D Acquisition
  - ⇒ Per es, laser scanning, fotogrammetria…
- ✓ Generazione procedurale
  - ⇒ Per es, modelli di piante o vegetali generati attraverso una simulazione della loro cresita
- ✓ Simulazione
  - ⇒ Per es, una simulazione fisica di un liquido per generare la sua superficie in movimento
- ✓ Modellazione manuale
  - ⇒ Da parte di artisti digitali

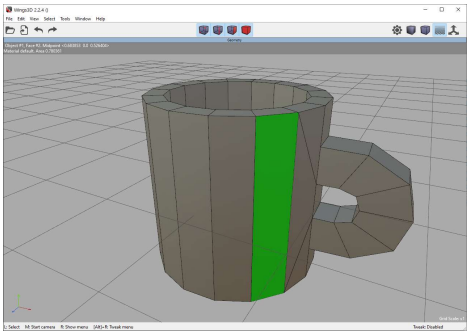
questo argomento

121

Generazione manuale di mesh (cenni)

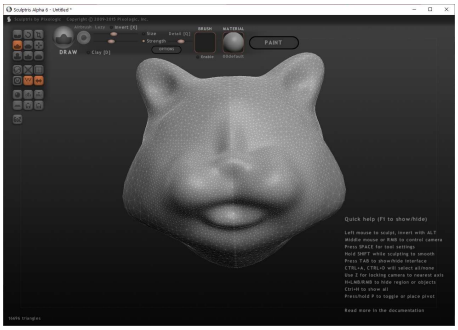
✓Due paradigmi di modellazione manuale:

Direct-poly modelling



esempio con Wings3D

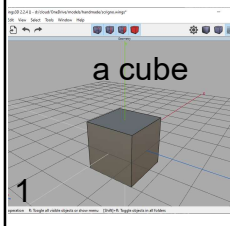
Digital Sculpting

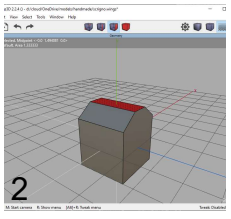


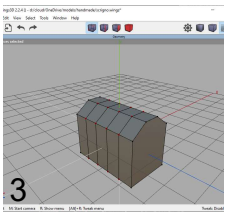
esempio con Sculptris Alpha

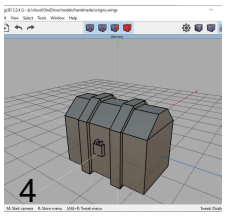
122

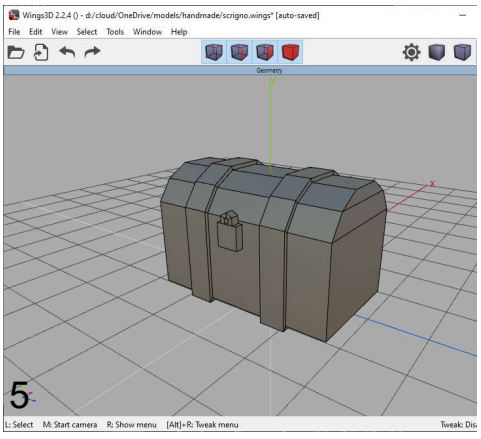
Direct Low-poly Modelling







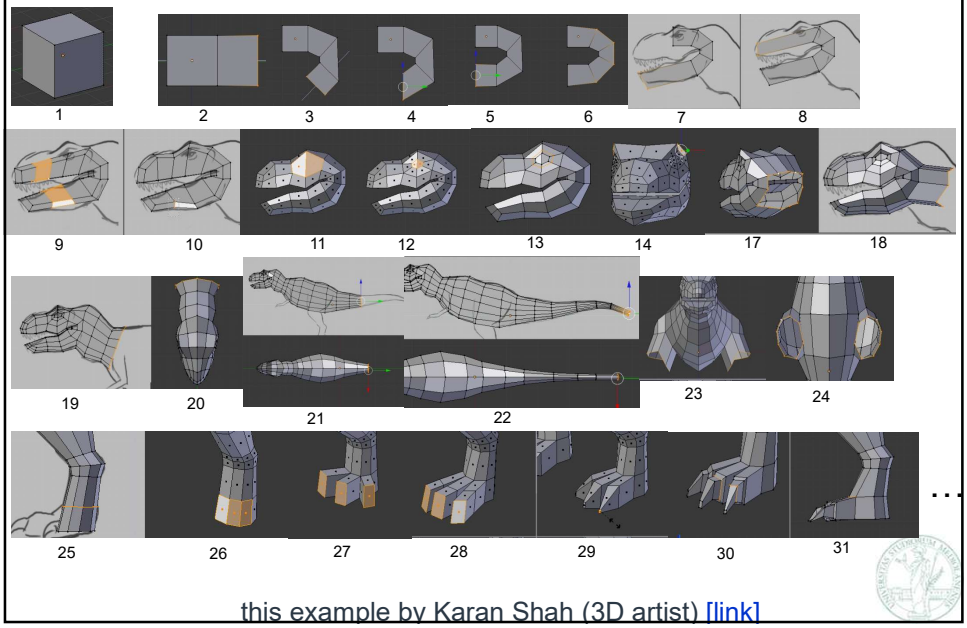




with wings3D

123

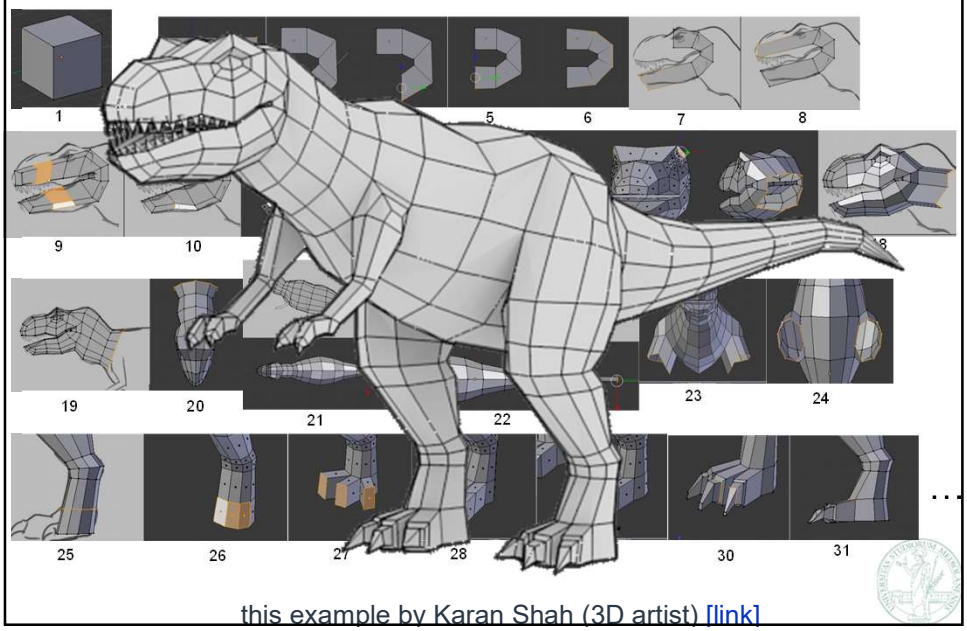
Direct Low-poly Modelling



this example by Karan Shah (3D artist) [link]

124

Direct Low-poly Modelling



this example by Karan Shah (3D artist) [link]

125

Direct (Low-poly) Modelling: note

- ✓ Il modellatore (un'artista 3D specializzato: il modellatore) manipola direttamente, attraverso un'apposita interattiva, gli elementi della mesh
  - ⇒ come facce, edge, vertici
  - ⇒ operazoni come: estrudere facce, connettere vertici, spostare vertici, suddividere edge o facce, dissolvere edge, etc
  - ⇒ Partendo da una forma semplice (es.: un cubo) si ottiene, dopo molti passaggi, la forma desiderata rappresentata nel modo opportuno
- ✓ Il modellatore edita esplicitamente
  - ⇒ La connettività della mesh (per es, quali edge connettono quali vertici)
  - ⇒ La posizione dei vertici (singolarmente, o a piccoli gruppi)
- ✓ Le operazioni mantengono automaticamente 2-manifoldness e (se serve) chiusura delle mesh
- ✓ Tipici risultati:
  - ⇒ mesh poligonali, spesso quad-dominant (o pure quad),
  - ⇒ spesso semi-regolari,
  - ⇒ spesso a media o bassa risoluzione (low poly)
- ✓ *Anticipazione:*
  vengono usati in questo contesto anche argomenti che vedremo: **superfici di suddivisione** e **superfici parametriche**

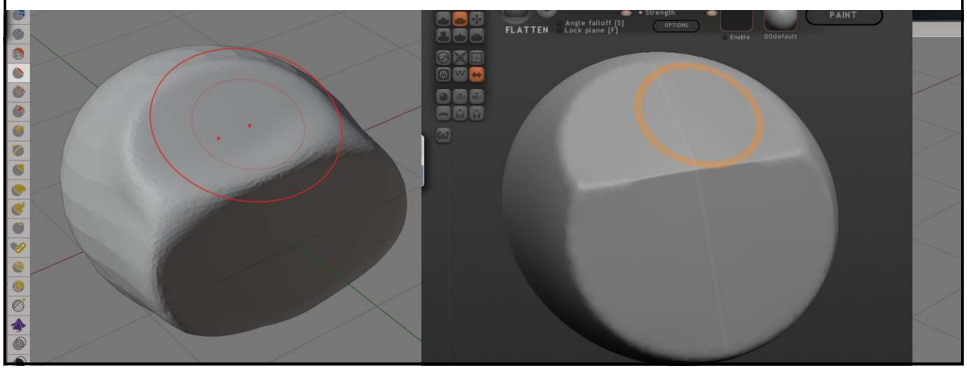
126

Digital Sculpting

mouse
(o stylus)

=

scalpello



127

Digital sculpting: note

- ✓ Il modellatore scolpisce la forma attraverso pennellate («brush strokes») che imitano la manipolazione di un materiale plasmabile come creta / pongo / etc
  - ⇒ Ma ovviamente in modo libero da ogni vincolo fisico: per es, è possibile rimuovere o creare materiale a piacere
- ✓ La rappresentazione della mesh (facce, vertici, edge) non è esposta al modellatore, ma viene gestita internamente
  - ⇒ Senza comunicarla necessariamente all'artista
- ✓ I «Pennelli» (anzi, scalpelli) digitali con effetti diversi, per es, local smoothing (rendere più liscio), appiattire, estrudere, «pizzicare», scavare, etc
- ✓ Il sistema gestisce il meshing automaticamente in background,
  - ⇒ mantenendo anche una una **risoluzione adattiva** in funzione della forma che viene generata
- ✓ Maggiormente adatta per modelli biologici / naturali
- ✓ Tipico risultato: una tri-mesh irregolare ad alta risoluzione

128

Authoring delle mesh (cioè 3D modelling)

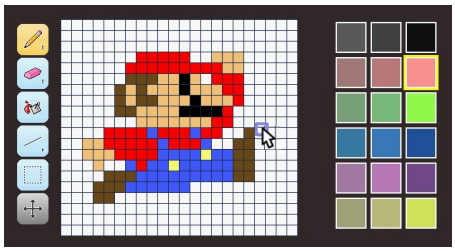
- ✓ Alcuni dei software di modellazione più diffusi

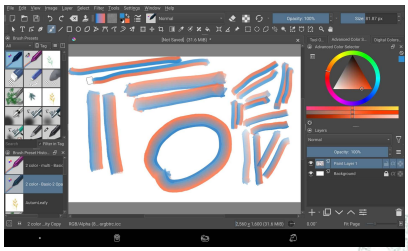
- ✓ **3D Studio Max** (autodesk) , **Maya** (autodesk) , **Cinema4D** (maxon) **Lightweight 3D** (NewTek), **Modo** (The Foundry) , **Blender** (open source!) ,
  - ⇒ all-purpose,completi
- ✓ **AutoCAD** (autodesk), **SolidWorks** (SolidThinking)
  - ⇒ Specializzati per il CAD
- ✓ **ZBrush** (pixologic), **Mudbox** (autodesk)
  - ⇒ Specializzati per lo scuplting
- ✓ **Wings3D**
  - ⇒ Solo low-poly modelling (& superfici di suddivisione) open-source, piccolo, specializzato
- ✓ **Sculptris Alpha**
  - ⇒ Solo sculpting open-source, piccolo, specializzato
- ✓ **[Rhinoceros]**
  - ⇒ parametric surfaces (NURBS)
- ✓ **FragMotion**
  - ⇒ small, specialized on animated meshes
- ✓ + molti altri con scopi più specifici
  - ⇒ editing of human models, of architectural interiors, environments, or specific editors for game-engines, etc...

129

Un'analogia col painting di immagini rasterizzate

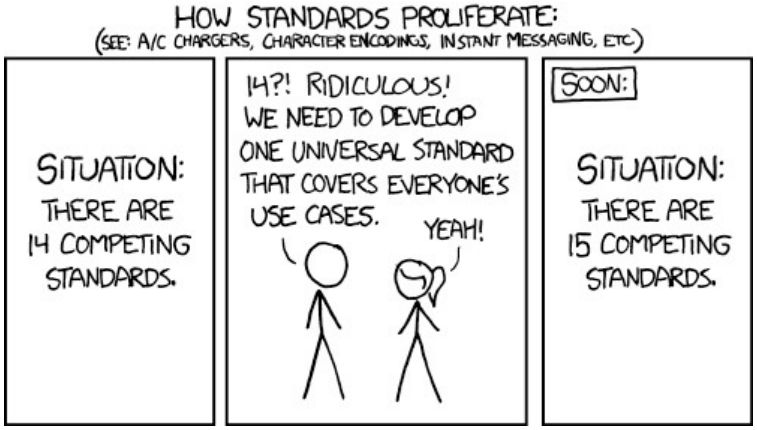
- ✓ Anche per l'editing di immagini rasterizzate (cioè costituite da matrici di pixel) esistono …
  - ⇒ approcci classici dove (come nel low-poly modelling) viene esposta la struttura interna dell'immagine: l'operatore umano manipola direttamente i pixel (vedi «pixel-art»)
  - ⇒ e approcci più recenti dove (come del digital sculpting) si utilizza una metafora (in questo caso, per es, pennelli e vernici) per consentire ad un operatore di generare l'immagine *prescindendo* dai pixel di cui è composta





130

Formati di file per mesh



(from **xkcd.com**)

131

Alcuni dei formati di interscambio mesh più diffusi

.OBJ (wavefront)
- ☺ Molto diffuso
- ☺ Indexed mesh
- ☺ ASCII: facile da leggere
- ☹ attributi per vertice: solo normali, UV-map (vedi poi)

.PLY (cyberware)
- ☺ customizzabile: qualsiasi attributo per vertice o per faccia
- ☹ abbastanza usato ma forse solo in ambito accademico

.FBX (AutoDesk)
- ☺ Abbastanza diffuso
- ☹ Propretario (non OpenSource)
- ☹ Supporto per mesh animate

. glTF (Chronos)
"Graphic Library Tranmission Format"
- ☺ molto completo e customizzabile
- ☺ open standard
- ☹ include animazioni, materiali, e molto altro

. DAE – COLLADA (Chronos)
In pratica versione precedente di glTF
- ☺ bastato su XML

132

Nella prossima lezione

- ✓ Come abbiamo visto, la **surface reconstruction** è un task di **geometry processing** il cui l'output è una **polygonal mesh**
- ✓ Vedremo alcuni altri tipici task del **geometry processing**, in cui la mesh è l'input
  - ⇒ spesso, anche l'output
- ✓ Molti di questi task possono essere visti come modi di conferire alle mesh proprietà desiderate in termini di quelle viste:
  - ⇒ Two-manifoldness
  - ⇒ Orientamento consistente
  - ⇒ Regolarità
  - ⇒ Risoluzione
  - ⇒ (compreso: risoluzione adattiva)
  - ⇒ Tipo dei poligoni (es: quad o triangoli)
  - ⇒ Chiusura
  - ⇒ Smoothness (continuità di normale)…
- ✓ Vedremo anche le stutture dati interne che consentono la manipolazione efficiente delle mesh

134