# Solid Modeling

http://hm.softalliance.net/

*Alexander Pasko, Evgenii Maltsev*

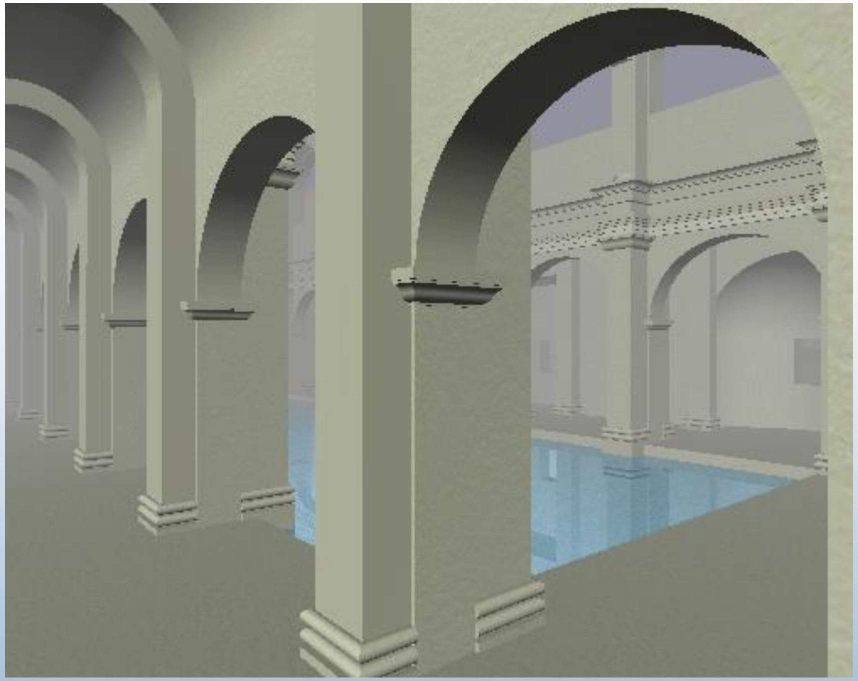


www.povcomp.com/hof/Villarceau_Circles-CSG.html

# Contents

1. Characterization of representation schemes for solids

- “Abstract solid”
- Definitions
- Formal properties of representations
- Informal properties



**The svLis model of the Great Bath in in Aquae Sulis as it was in 200 AD**

# Contents

2. Schemes for representing solids

- Ambiguous schemes
- Pure primitive instancing
- Cell decompositions
- Spatial occupancy enumeration
- Constructive Solid Geometry
- Sweep representations
- Boundary representations
- Medial Axis Transforms

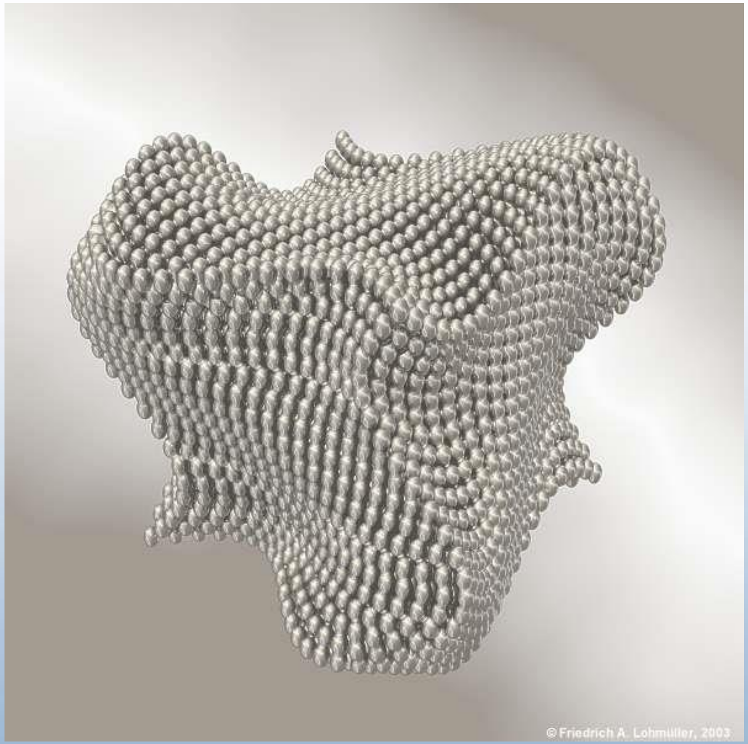
*"Still with Bolts" by Jaime Vives Piqueres (2002)*

# Contents

- Ray representations
- Parametric function representations
- Real function representations
- Hybrid schemes and conversions

3. Solid modelling systems

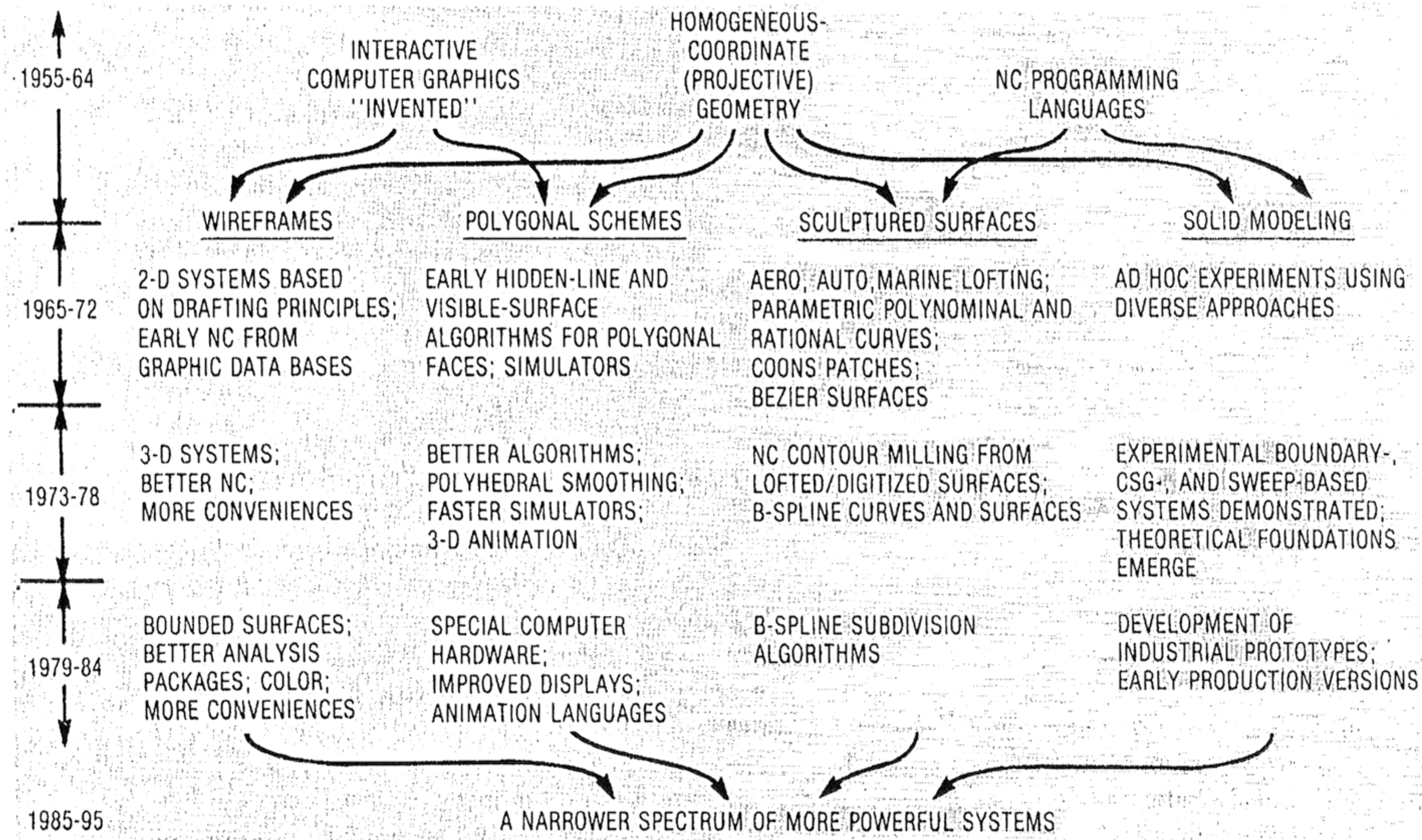
4. References




"Dancing Cube" by Friedrich A. Lohmueller (2003)

A. A. G. Requicha, H. B. Voelcker, 1982

A historical summary of approaches to 3-D object representation

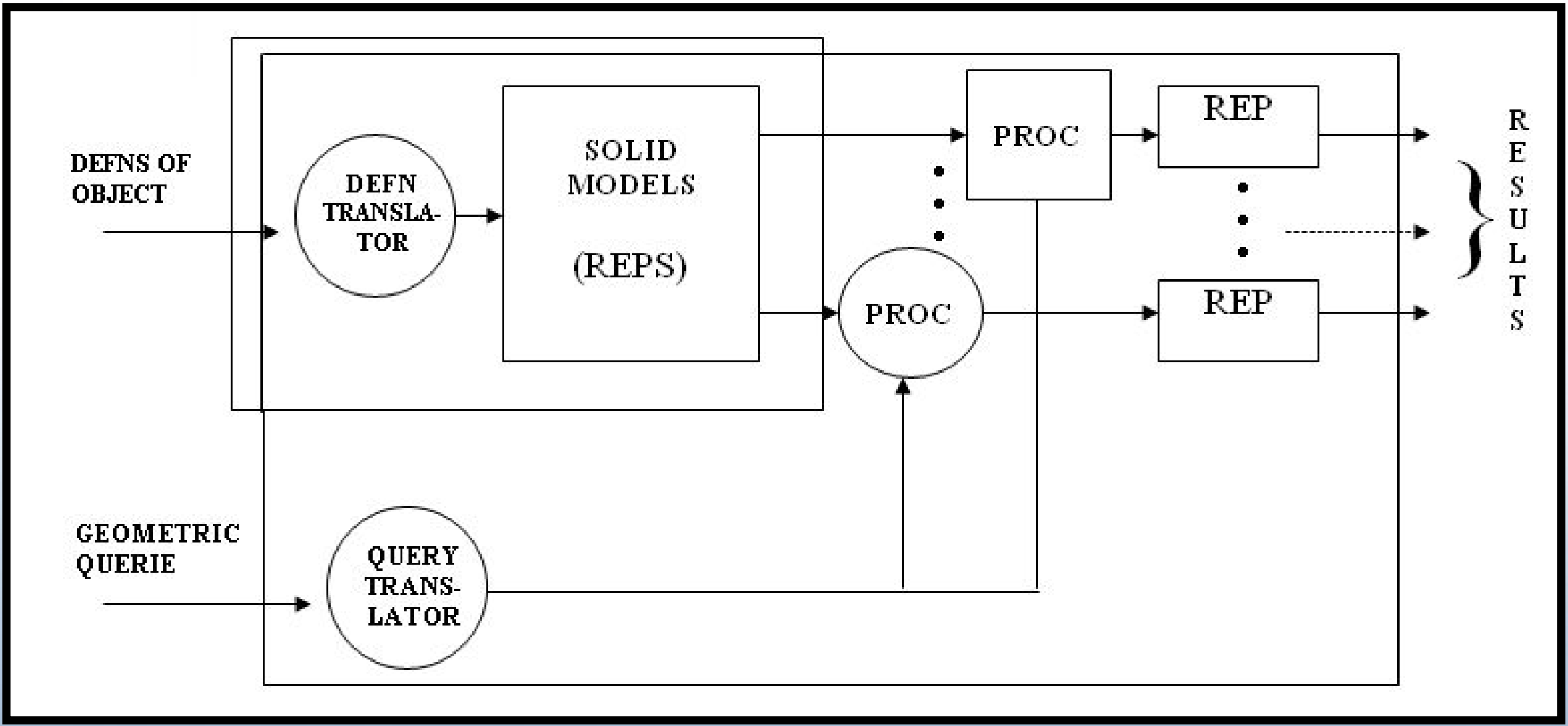


# Solid Modelling System

A solid modelling system has four primary components [Requicha, 1980]:

- Symbol structures which represent solid objects;
- Processes which answer geometric questions (such as “What is the volume?”) using the symbol structures;
- Input facilities for creating and editing object representation;
- Output facilities and representations of result.

# Solid Modelling System




Requicha, Comp. Surveys, p.438

Solid modelling system is a subsystem which provides entering, storing and modifying object representation.

# Abstract solid

Properties that should be captured by the notion of "abstract solid":

- Rigidity: An abstract solid must have an invariant shape which is independent of the solid's location and orientation.
- Homogeneous three dimensionality: A solid must have an interior, and a solid's boundary cannot have isolated or dangling portions.
- Finiteness: A solid must occupy a finite portion of space.

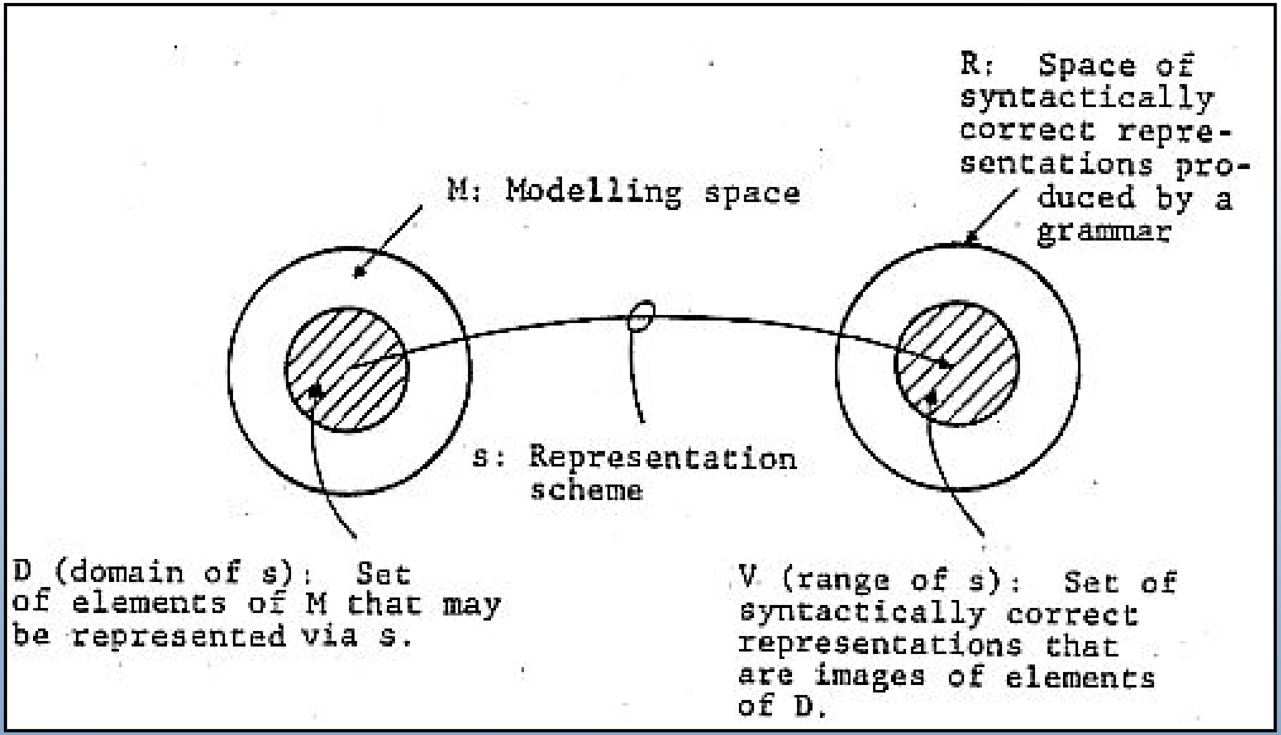
# Abstract solid

- Closure: Rigid motions (translations and rotations) or operations that add or remove material (set-theoretic operations) must produce another solid.
- Finite describability: There must be some finite aspect of 3D models of solids (a finite number of “faces”) to make them representable in computers.
- Boundary determinism: The boundary of a solid must determine unambiguously what is “inside” the solid.

A representation scheme establishes a correspondence between *M* and *R*.

It is defined as relation

s: M → R




Requicha, Comp. Surveys, p.441

# Abstract solid : r-sets

Suitable models for solids are r-sets that are bounded, closed, regular, and semi analytic subsets of 3D Euclidean space ($E^3$).

- A bounded set

  occupies a finite portion of space.

- Closed set

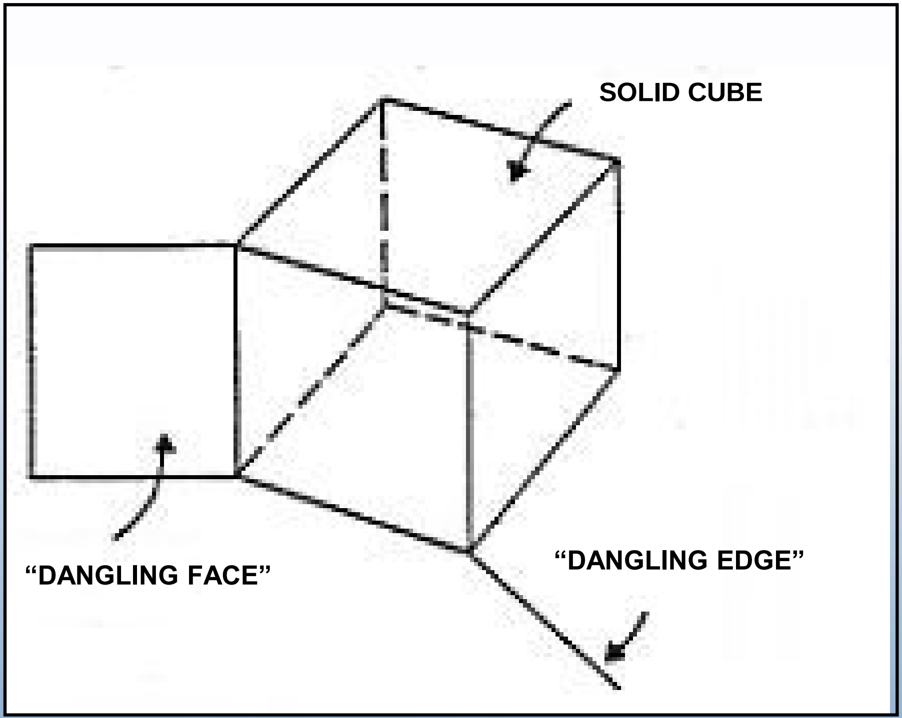
  The set is closed if it contains its boundary.

# Abstract solid : r-sets

- Reqular

It is a closed set. It contains its boundary.

It is not a regular set because its boundary has dangling portions that are not adjacent to the set's interior.

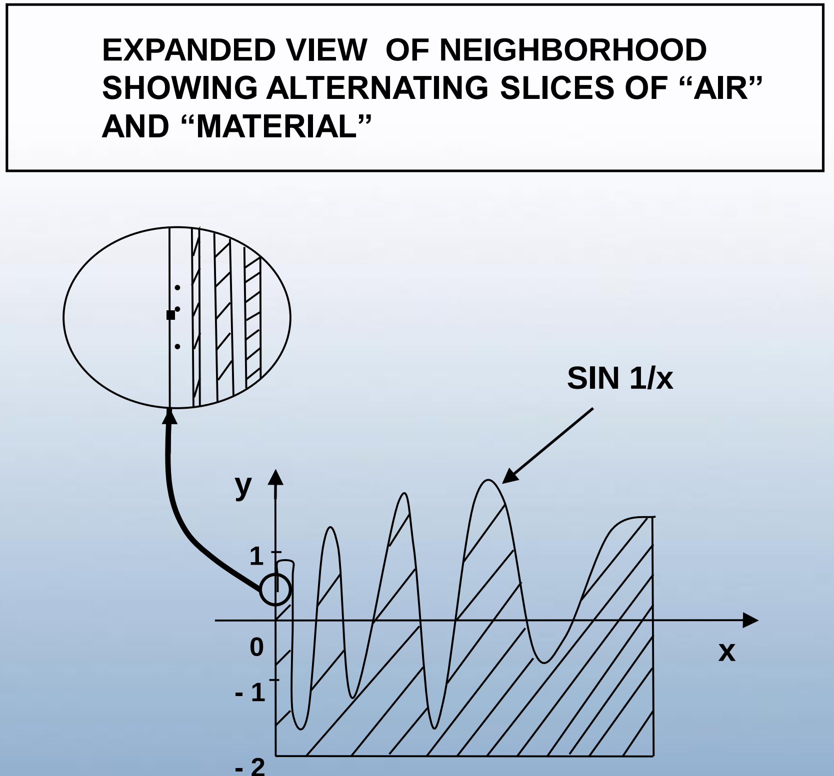



Requicha, Comp. Surveys, p.440

# Abstract solid : r-sets

- Semi-analytic set

This set is not semi-analytic because its top face is ill-behaved; it oscillates infinitely as it approaches the left face.




Requicha, Comp. Surveys, p.440

# Representation scheme

Syntactically correct representations are finite symbol structures constructed according to syntactical rules.

- The collection of all syntactically correct representations is called a representation space *R*.

- Abstract solids (r-sets) are the elements of a mathematical modelling space M.

# Formal properties of representations

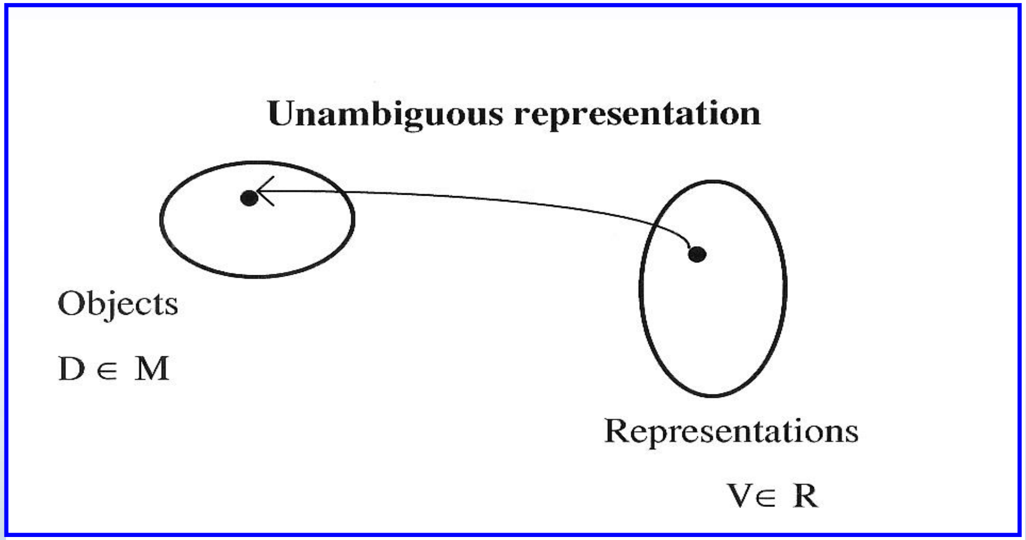
- Domain

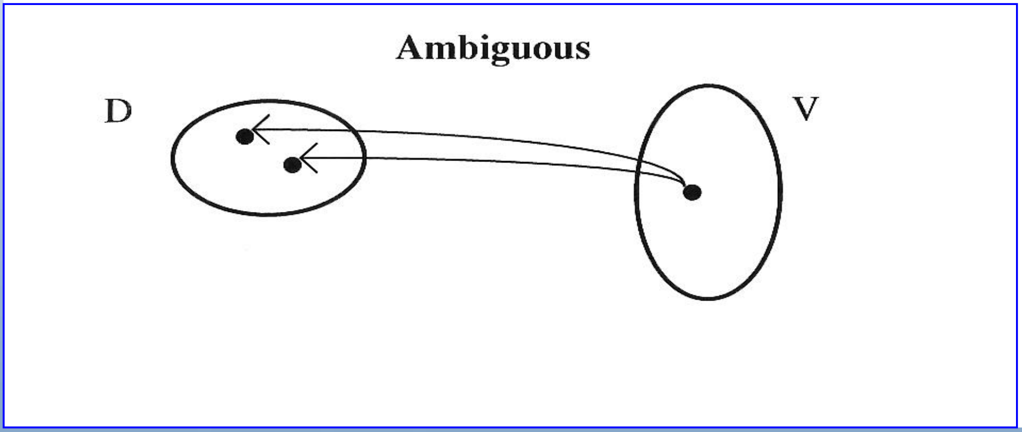
  The domain is the set of solids representable in the representation scheme (its descriptive power).

- Validity

  A symbol structure, which corresponds to a nonsense object, should not exist.

# Formal properties of representations



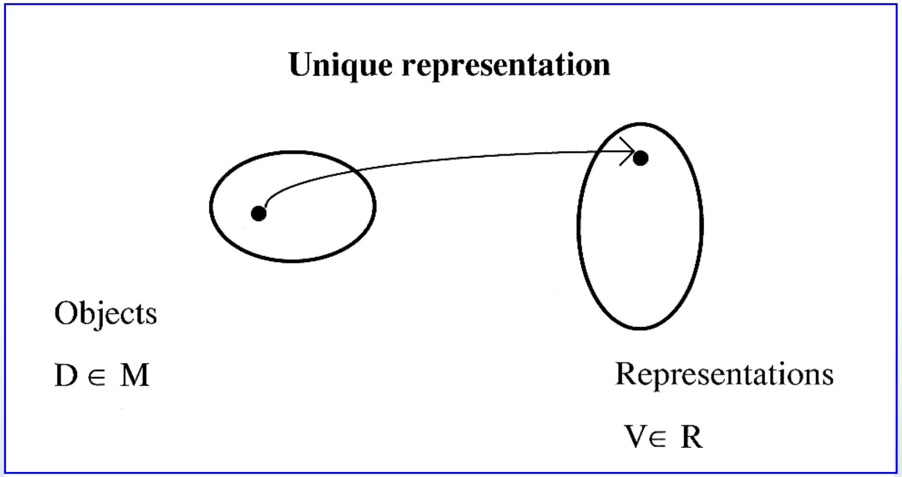

- Completeness ( non-ambiguity)

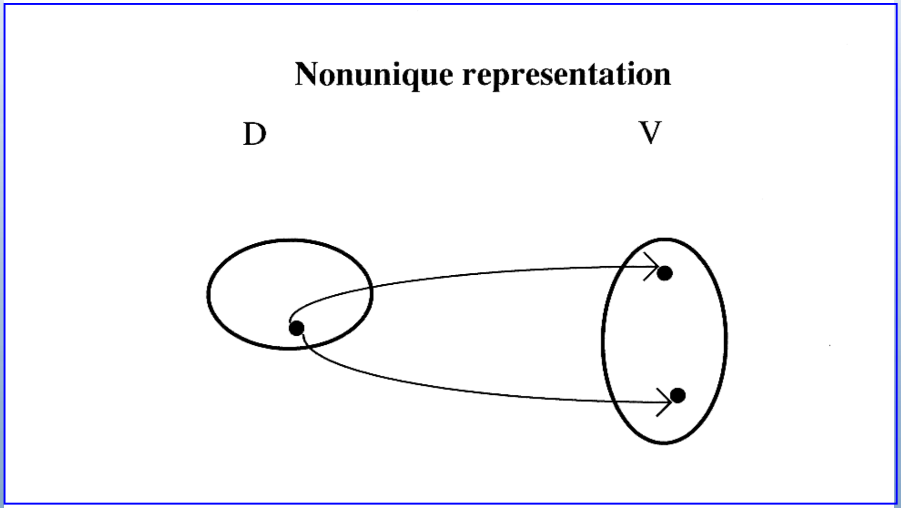
A representation $r \in V$ is unambiguous if it corresponds to a *single object in D*. A scheme is complete or unambiguous if all of its valid representations are unambiguous.

# Formal properties of representations

- Uniqueness. A scheme is unique if for one object there is one and only one representation.

Representational schemes which are both unambiguous (complete) and unique are highly desirable because they are one-to-one mappings.



# Informal properties of representation schemes

- Conciseness.

  This refers to the “size” of representations in a scheme (proportional to required memory). Compare: equation and polygonal model.

- Ease of creation.

  Concise representations generally are easier to create. Input subsystems are needed to help users to create representations.

# Informal properties of representation schemes

- Suitability for applications.

Example: Roman numbers are not convenient for arithmetic operations. In solid modelling, no single representation is uniformly “best”.

Multiple representations are suitable for general-purpose solid modelling systems.

# Ambiguous schemes

1) Engineering drawings (drafts).

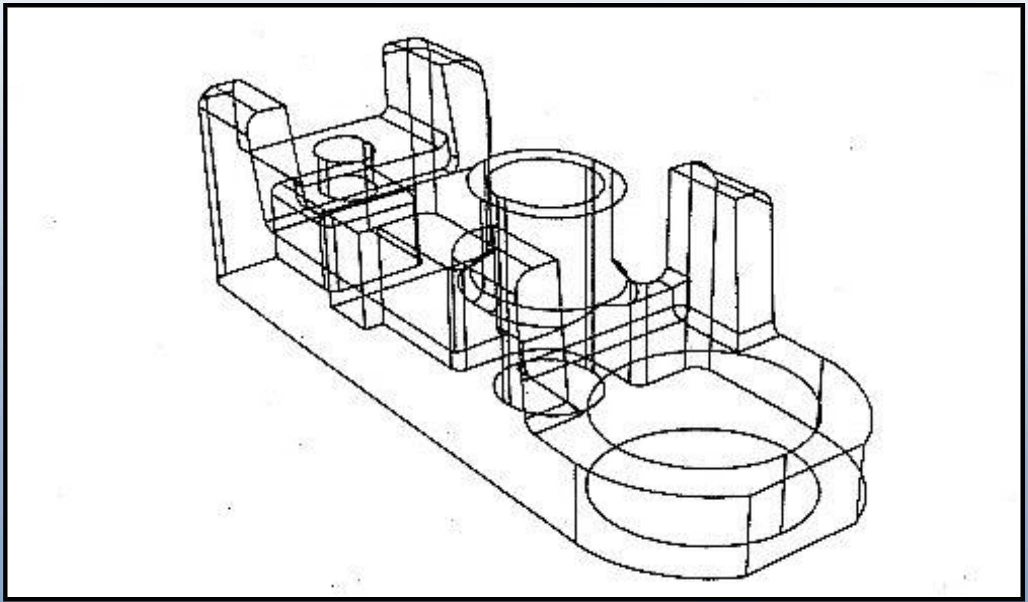
No formal definition as a representation scheme.

More formal corresponding schemes:

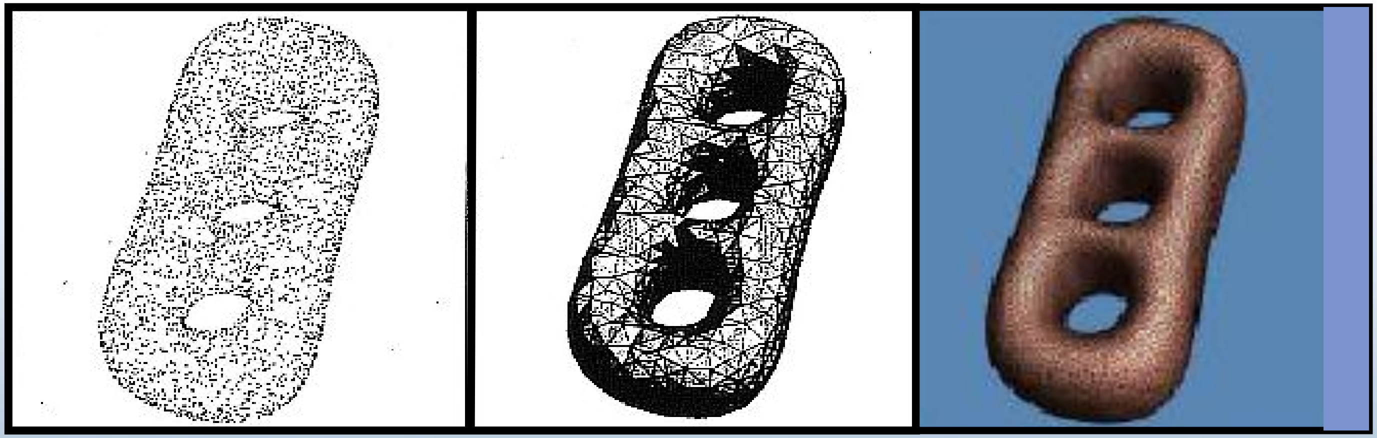
- Collections of planar projections.

  Mapping to a 3 solid is needed.

- Suitable collections of 3D entities. Selection of “edges” leads to wire frame representations.



## 2) Measurements of physical solids.



(a) Input: 4000 unorganized points P

(b) Step 1: Reconstructed mesh M

(c) Step 2: Smoothed mesh

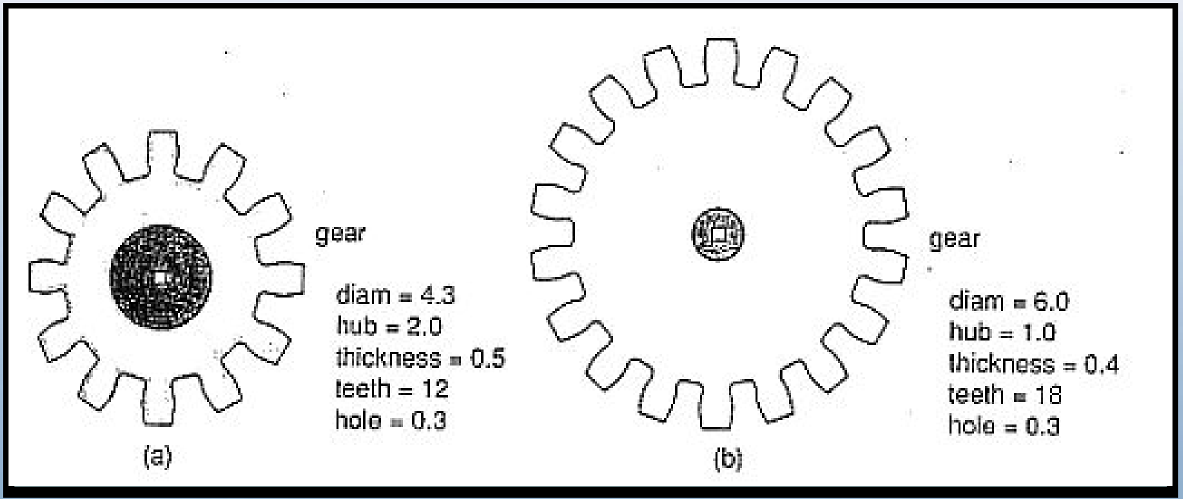
A solid is represented by a set of coordinates of points lying on the boundary or inside the object.

# Pure Primitive Instancing

The modelling system defines a set of primitive 3D solid shapes for the specific application area:

$$\mathbf{primitive}_i\,(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k)$$

- Primitive with parameters define a family of parts;
- Primitives may include complex objects (gears, bolts, etc.);




Two gears defined by primitive instancing.

- No operations to form a new more complex object;
- Only one way to create a new kind of object - to write the code that defines it;
- Programs to draw or to calculate mass properties must be written individually for every primitive.

# Pure primitive instancing

Properties:

- Unambiguous.

  A set of parameters defines one solid.

- Unique.

  For a solid only one set of parameters exists.

- Concise and easy to validate.

# Properties

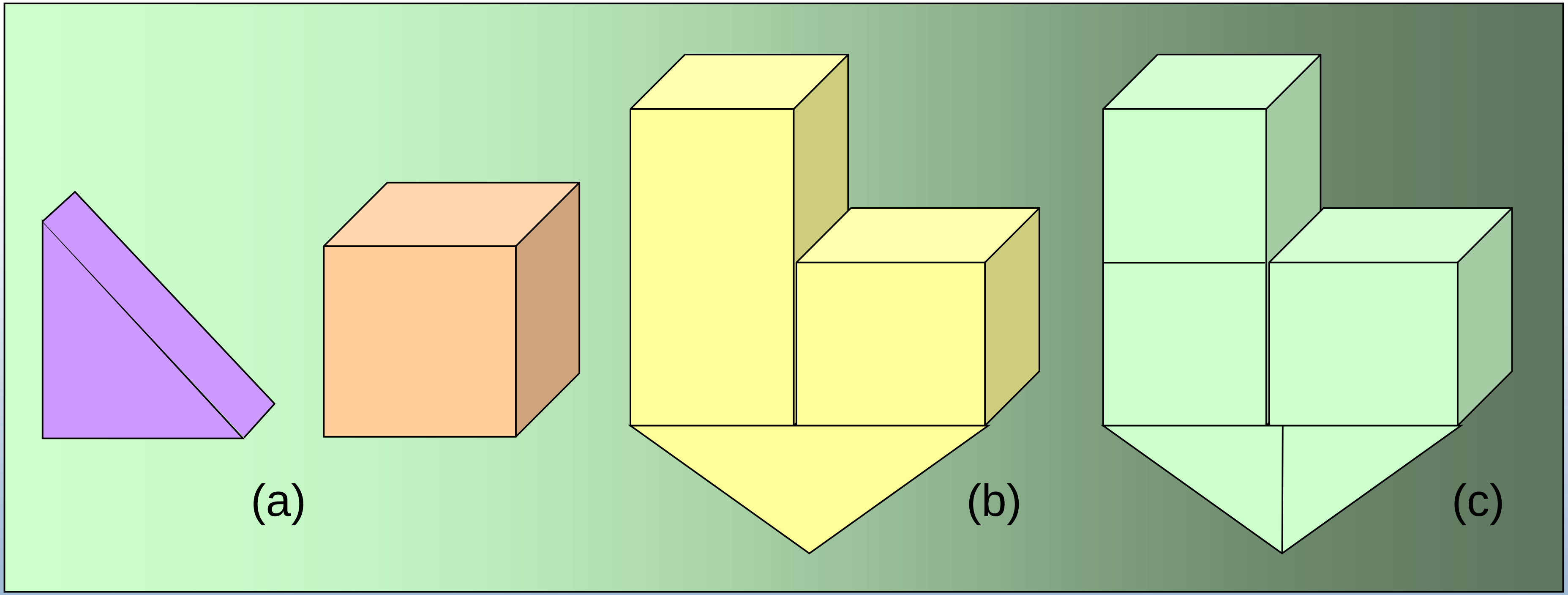
- Easy to use.
- Domains are small enough to be covered by a small catalog of primitives with small number of parameters.
- Efficient in specific applications but allow no uniform treatment.

# Cell decompositions

- Parameterized set of primitive cells that are often curved
- Constructing complex object by “gluing” primitive cells together
- Restrictions on the “glue” operation often require that two cells share a single point, edge, or face

# Cell decompositions

- Representation is not unique:




The cell shown in (a) may be transformed to construct the same object shown in (b) and (c) in different ways.

# Properties

- Unambiguous

  A set of cells defines one solid

- Validity

  is computationally expensive to establish

- Not concise

- Not easy to create

# Properties

- Convenient

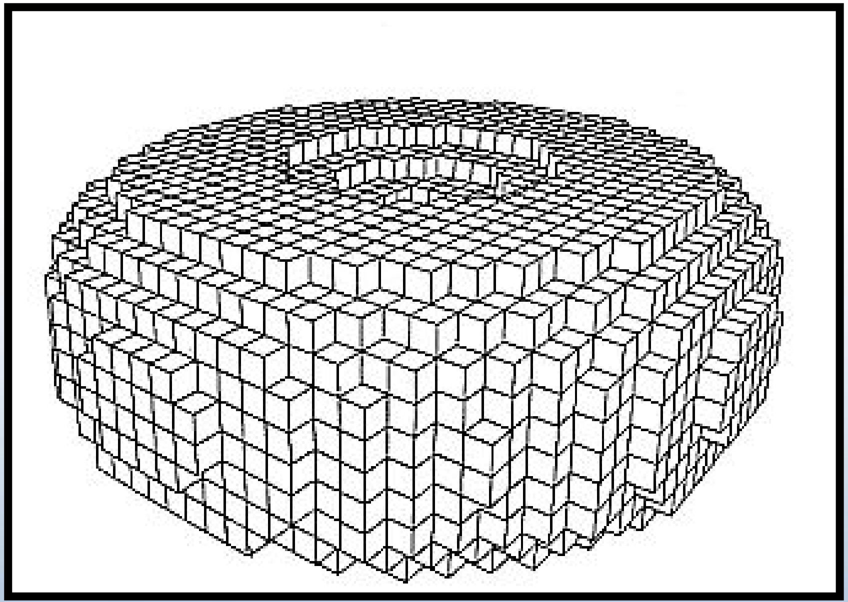
for computing topological properties:

- "one-piece" detection (connectivity)
- check "voids" or "holes"

Cell decomposition is used in 3D finite element methods (FEM) for the numerical solutions of differential equations.

# Spatial occupancy enumeration

- Special case of cell decomposition with identical cells arranged in a fixed, regular grid
- The cells are often called voxels
- The most common cell type is the cube, and the representation of space as a regular array of cubes is called a voxel array (a cuberrile)



Torus represented by spatial-occupancy enumeration.

# Spatial occupancy enumeration

- For every cell, only its presence or absence in the grid is defined
- A cell is presented in the grid if it is occupied by the object
- Disadvantages:

  - approximate model, no concept of “partial occupancy”
  - memory consuming (up to n3 cells)

# Properties

- Unambiguous.

  A set of voxels defines one solid.

- Unique.

  For a solid only one set of voxels with the given grid step exists.

# Properties

- Easy to validate.
- Not concise (verbose).
- Efficient in applications where objects are boxlike (architecture) or extremely irregular (biomedical).

# Contents

## 2. Schemes for representing solids

- Ambiguous schemes
- Pure primitive instancing
- Cell decompositions
- Spatial occupancy enumeration
- **Constructive Solid Geometry**
- Sweep representations
- Boundary representations
- Medial Axis Transforms

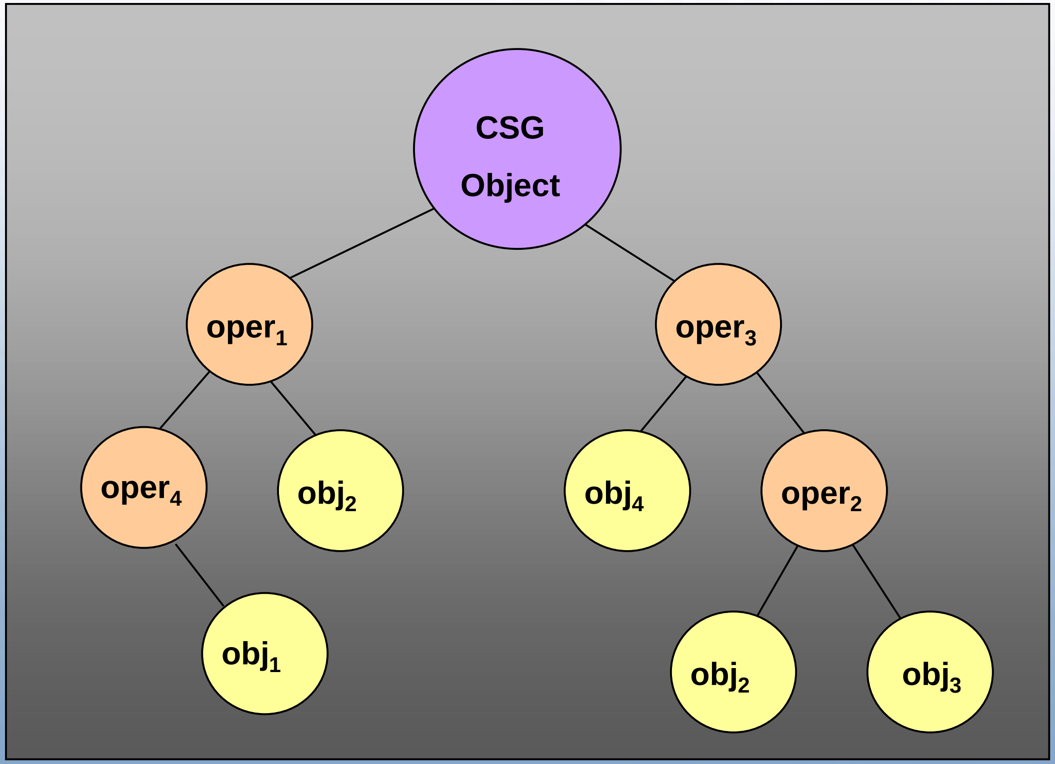
*"Still with Bolts" by Jaime Vives Piqueres (2002)*

# Constructive Solid Geometry (CSG)

- CSG is based on a set of 3D solid primitives and regularized set-theoretic operations
- Traditional primitives: block, cylinder, cone, sphere, torus
- Operations; union, intersection, difference + translation and rotation

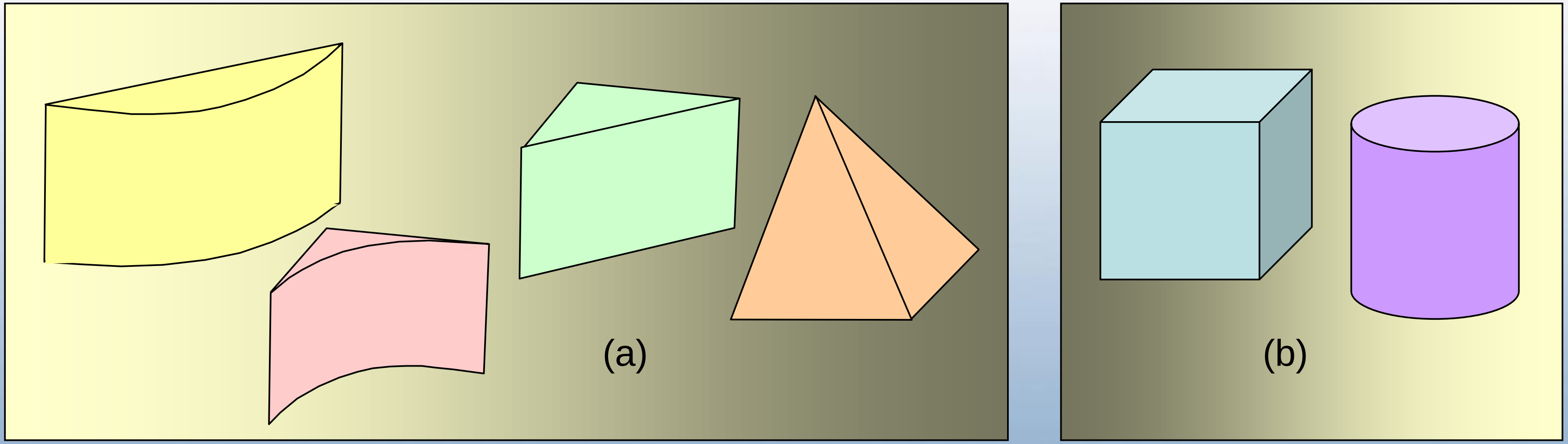
# CSG tree

- A complex solid is represented with a binary tree usually called CSG tree



# Properties

- Domain depends on the set of primitives and on the set of operations.




Two CSG schemes having different primitives but the same domain.

# Properties

- Unambiguous

  A CSG tree defines one solid.

- Nonunique

  There are several possible CSG trees for one solid.

- Validity

  Any syntactically correct CSG tree is **valid**, if the primitives are r-sets.

  - CSG tree based on unbounded primitives may represent bounded sets and therefore be invalid.

# Properties

- Concise

  if primitives are well matched to the domain

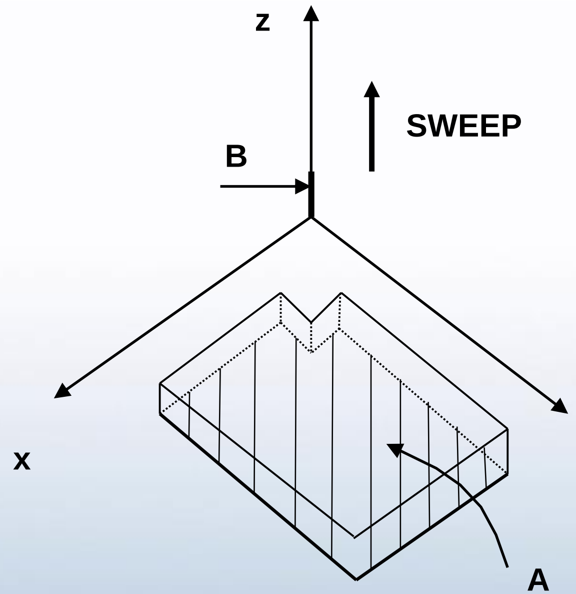
- Humans can easily create CSG representations

- Efficient

  for rendering and computing integral properties; not efficient for line drawings and certain types of graphic interactions (“pick an edge”).

# Sweep representation

- A set of all points visited by an object A moving along a trajectory B is a new solid, called a sweep.
- Translational sweeping (extrusion): 2D area moves along a line normal to the plane of the area.



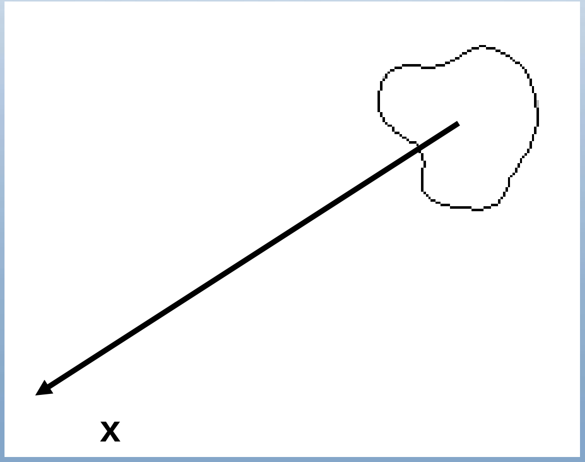
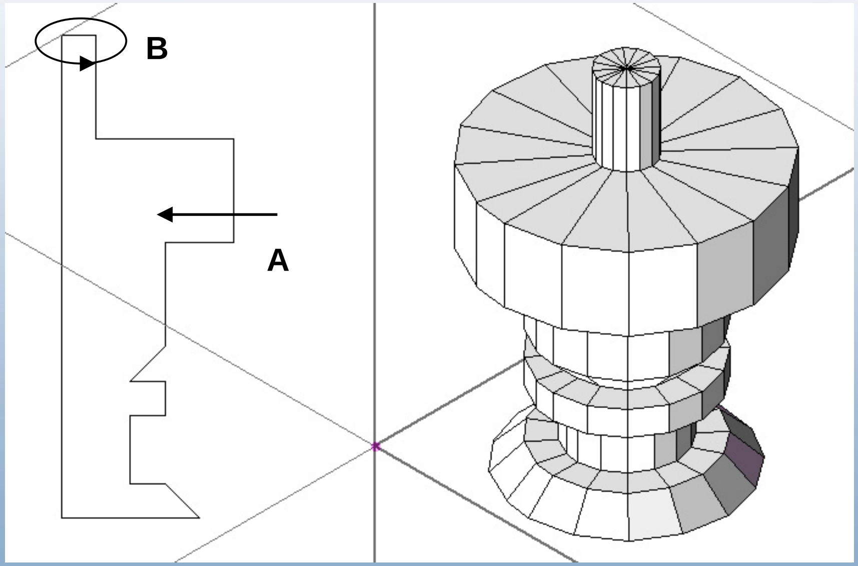




Image by Martin Culpepper, 1999

- Rotational sweeping is defined by rotating an area about an axis




www.tipus.uniroma3.it

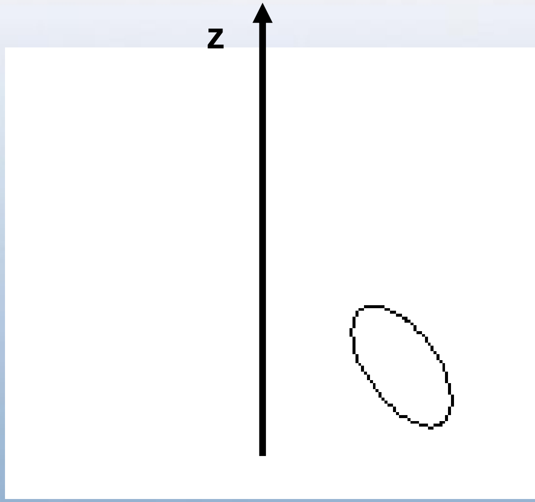



Image by Martin Culpepper, 1999

# Sweep representation

- Sweeps with a generating area changing in size, shape or orientation and following an arbitrary curved trajectory are called generalized cylinders.

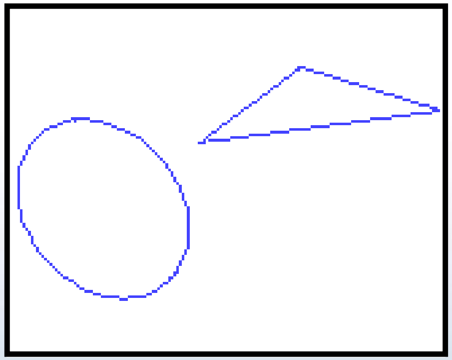


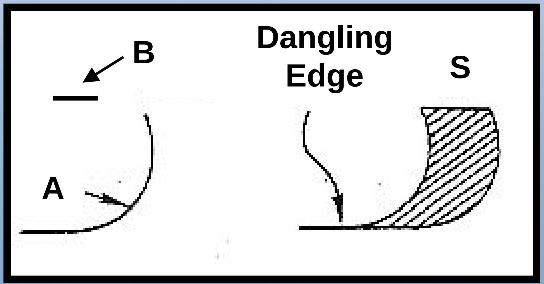
Image by Martin Culpepper

Problems: sweeping by moving solid, self-intersections, CSG operations on sweep.

# Properties

- Unambiguous

  A moving object + a trajectory define one solid
- Not unique
- General validity conditions for sweep representations are unknown. General sweeping may produce non-regular sets.



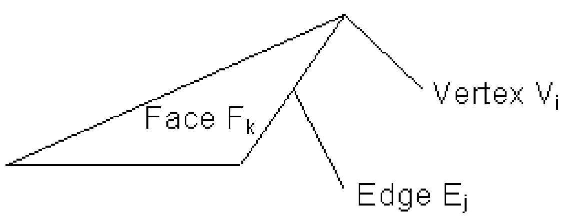

Requicha, Comp. Surveys, p.451

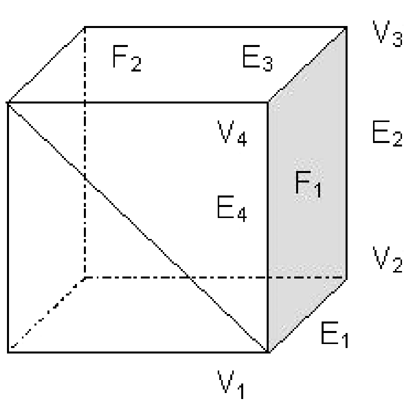
- Applications of sweeping by a moving solid: material removal (NC machining), dynamic interference of solids.

- Sweeping by a moving solid: lack of known algorithms for computing properties.

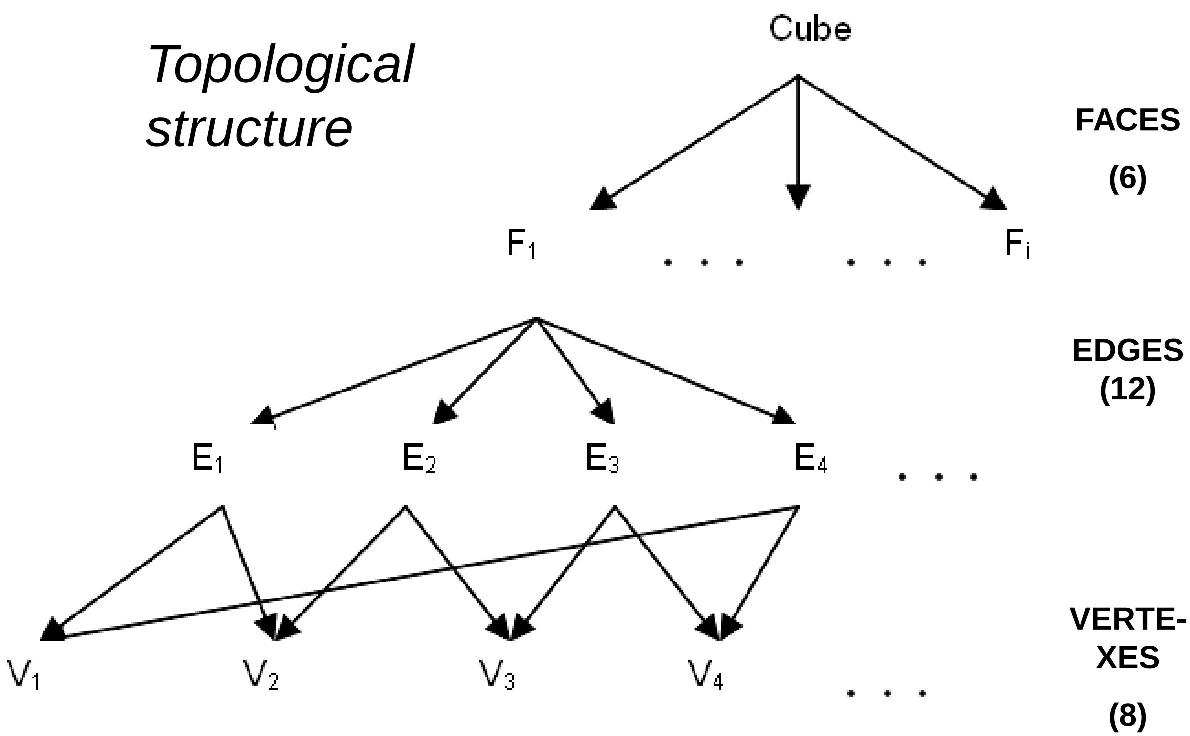
# Boundary representation

Example: A boundary representation for a cube




*Topological structure*



# Properties

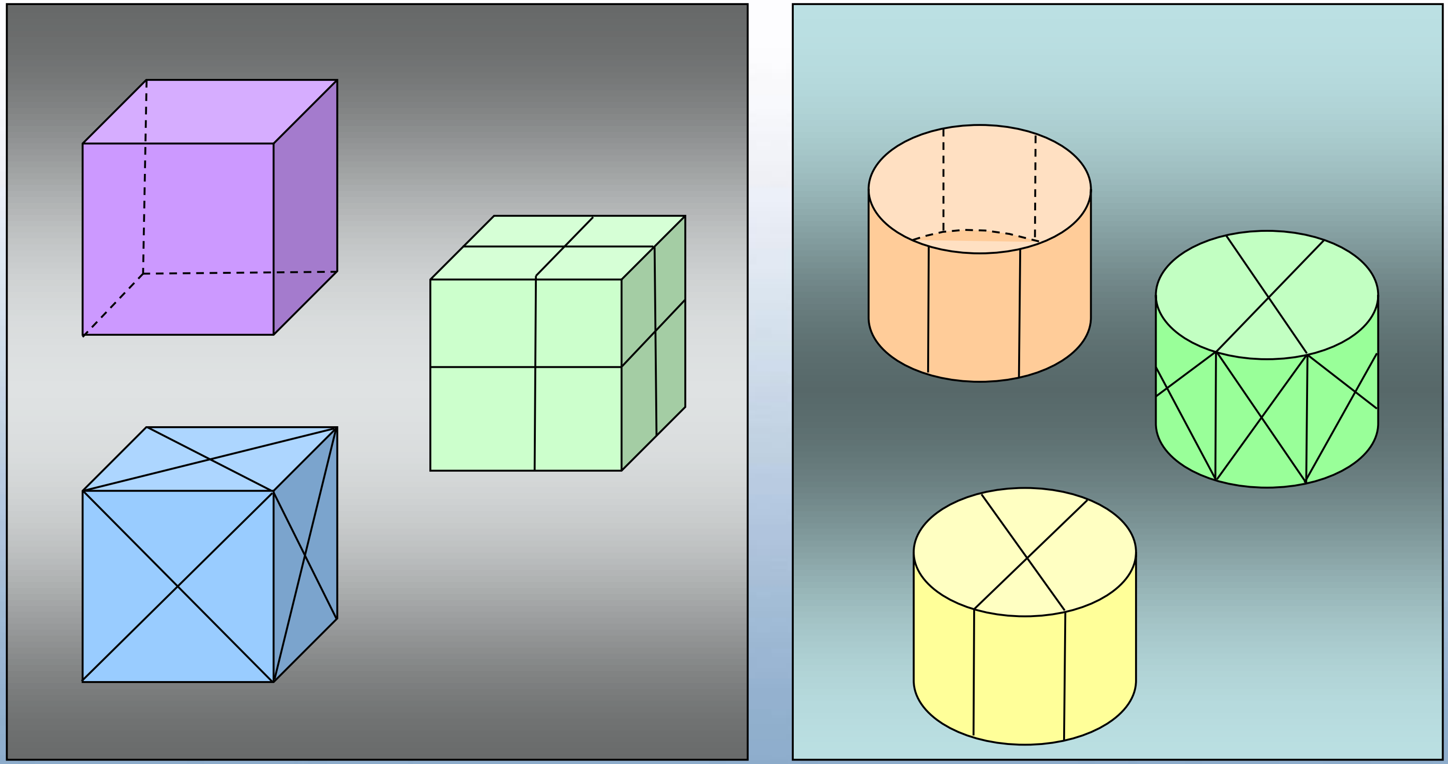
- Domains

are as reach as those of cell-decomposition or CSG schemes. Given a CSG scheme it is always possible to design BRep scheme with the same domain.

- Unambiguous

if faces are represented unambiguously.

- Not unique



- Validity
  control requires expensive calculations.
- Not concise (verbose)
  More than 10 times longer than corresponding CSG.
- Difficult
  for humans to construct.
- Efficient
  in line and shaded drawings, graphic interaction and topological applications.

ARISTIDES A. G. REQUICHA *Representations for Rigid Solids*

## SUMMARY OF REPRESENTATION SCHEME PROPERTIES AND APPLICABILITY

| Schemes | | PROPERTIES | | | | | | APPLICATIONS | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | DOMAIN | VALIDITY | COMPLETENESS | UNIQUENESS | CONCISENESS | EASE OF CREATION | NULL OBJECT | OBJECT EQUALITY | LINE DRAWINGS | GRAPHIC INTERACTION | SHADED DRAWINGS | MASS PROPERTIES | HOMOLOGICAL PROPERTIES | FINITE ELEMENT ANALYSIS | INTERFERENCE ANALYSIS | ROUGH MACHINING | FINISH MACHINING | NC TAPE VERIFICATION |
| PRIMITIVE INSTANCING | | | G | G | G | G | G | G | G | G | | ? | G | G | ? | | G | G | G |
| SPATIAL ENUMERATION | | | G | G | G | | | G | G | ? | ? | ? | G | G | G | G | | | |
| CELL DECOMPOSITION | | G | | G | | | | G | | ? | ? | ? | P | G | P | ? | ? | ? | ? |
| CSG | GENERAL 1/2 SPACES | G | | G | | | | ? | ? | | P | ? | ? | | | P | P | ? | P |
| CSG | BOUNDED PRIMITIVES | G | G | G | | G | G | ? | ? | | P | ? | ? | | | P | P | ? | P |
| SWEEP | TRANSLATIONAL & ROTATIONAL | | G | G | ? | G | G | G | ? | G | G | ? | G | ? | P | | | | |
| SWEEP | GENERAL | G | | G | | | | G | | | | | | | | | | | |
| BOUNDARY | | G | | G | ? | | | G | ? | G | G | P | ? | P | | | | P | |

KEY: G = GOOD P = PROMISING ? = UNCLEAR BLANK= UNPROMISING OR POOR

H. Voelcker, 1995

# *A Taxonomy of Complete Representation Schemes for Solids*

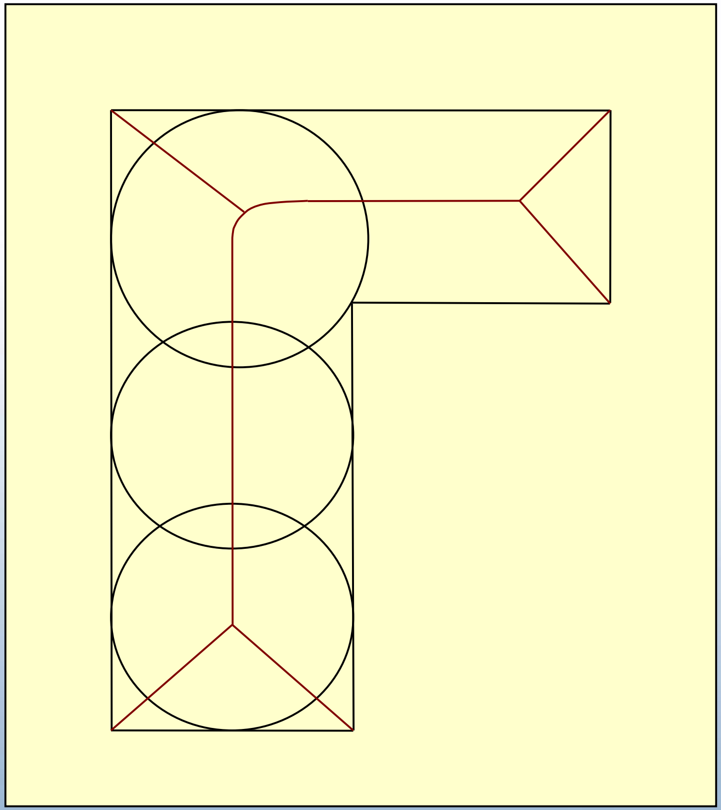
< H. Voelcker: 9/93; rev. 2/94, 2/95, 6/95 >

| PARADIGM | FAMILY NAME | VARIANT | MATHEMATICAL MECHANISM | ACCESSIBLE ATTRIBUTES | SPECIAL PROPERTIES |
|---|---|---|---|---|---|
| Instantiation | Templates[1] | | Special-Case Formulae | | |
| Cell Composition (Gluing) | Spatial Enumeration[2] | Linear subdivision of space ⇒ rectilinear grids of equal-size cells | Union of Quasi-Disjoint Solid Cells<br>$Solid = \cup C_i$<br>where<br>$C_i \cap_n C_j = \varnothing$, i _j | Interior, Discretized into Cells | Spatial Addressability |
| | | Recursive subdivision ⇒ hierarchically graded cells (quadtrees, octrees) | | | |
| | Cell Decomposition | Simplicial or Cell Complexes; elements produced by object partitioning (e.g. triangulation) | | | |
| | Boundary Representations (B-reps) | Exact boundaries | Union of Boundary Cells<br>$\partial Solid = \cup Face_i$<br>$\partial Face = \cup Edge_i$<br>(Solids thru boundary determinism) | Boundary; Surface 'Features' | |
| | | Linear approximations ('tilings', 'facets') | | | |
| Boolean Composition | Constructive Solid Geometry (CSG) | Halfspace primitives | Regularized boolean composition | Solid primitives; syntactic structure | Recursive partitioning |
| | | Bounded solid primitives | | | |
| Generation | Sweep Representations | Simple sweeps: S constant, M a single rotation or translation | 'Infinite Union':<br>Sweep(S, M(t)) =<br>$\cup S(t_i)$ @ $M(t_i)$, $\forall t_i \in$ [interval] | Generator Elements (S, M) | (Limited domains) |
| | | General sweeps | | | |
| | Medial Axis Transforms (MATs)[3] | | $\cup$ Spine(p) $\oplus$ Sphere(R(p)) | Spine (skeleton); Radius function | |
| Spatial Sampling | Ray Representations[3] | | Induced cell decompositions | Ray-sampled interior | Directional sampling |
| | Parametric Sample Sets[3] | | Interpolation | Sampled boundary | |
| Real and Vector Functions | Parametric-function Representations[3] | | Homeomorphic mapping over finite domains | Interior | (Limited domains) |
| | Real-function Representations[3] | Differentiable Indicator Functions | $f(p) > 0 \Rightarrow p \in iS$<br>$f(p) = 0 \Rightarrow p \in$ _S<br>$f(p) < 0 \Rightarrow p \in cS$ | Point classifications | Boolean composition through arithmetic operations |

[1] Called 'Pure Primitive Instancing' in Requicha's paper in *ACM Computing Surveys*, December 1980.
[2] Called 'Spatial Occupancy Enumeration', without linear and hierarchical distinctions, in Requicha's 1980 paper.
[3] Not acknowledged in Requicha's 1980 paper.

# Medial axis representations

The medial axis of a 2D object is defined as the closure of the locus of centers of maximal inscribed disks. A disk is maximal if no other disk contains it.

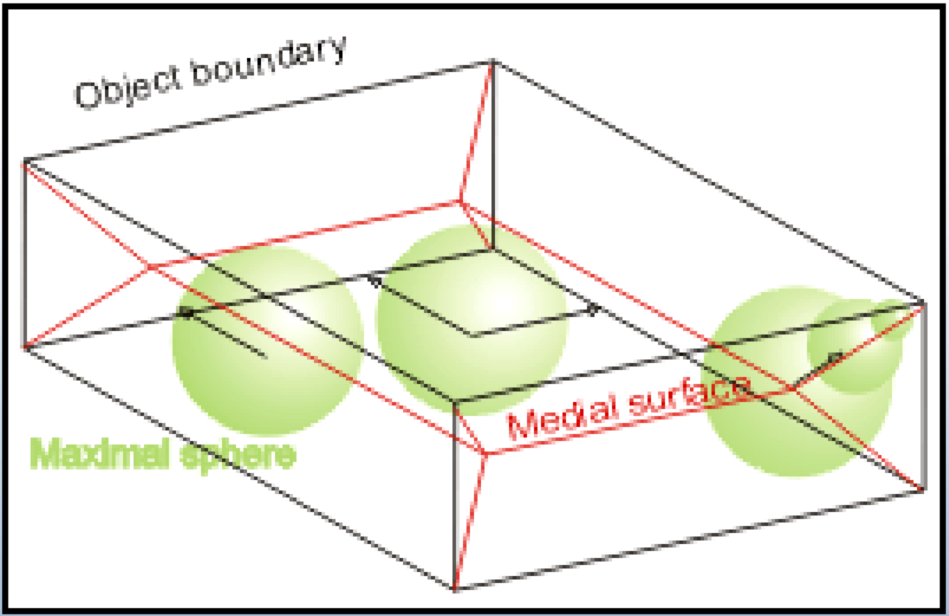


Hoffmann “Solid modelling” Overview

L-shaped domain and associated medial axis. Some maximal inscribed circles contributing to the medial axes are also shown.

# Medial axis representations

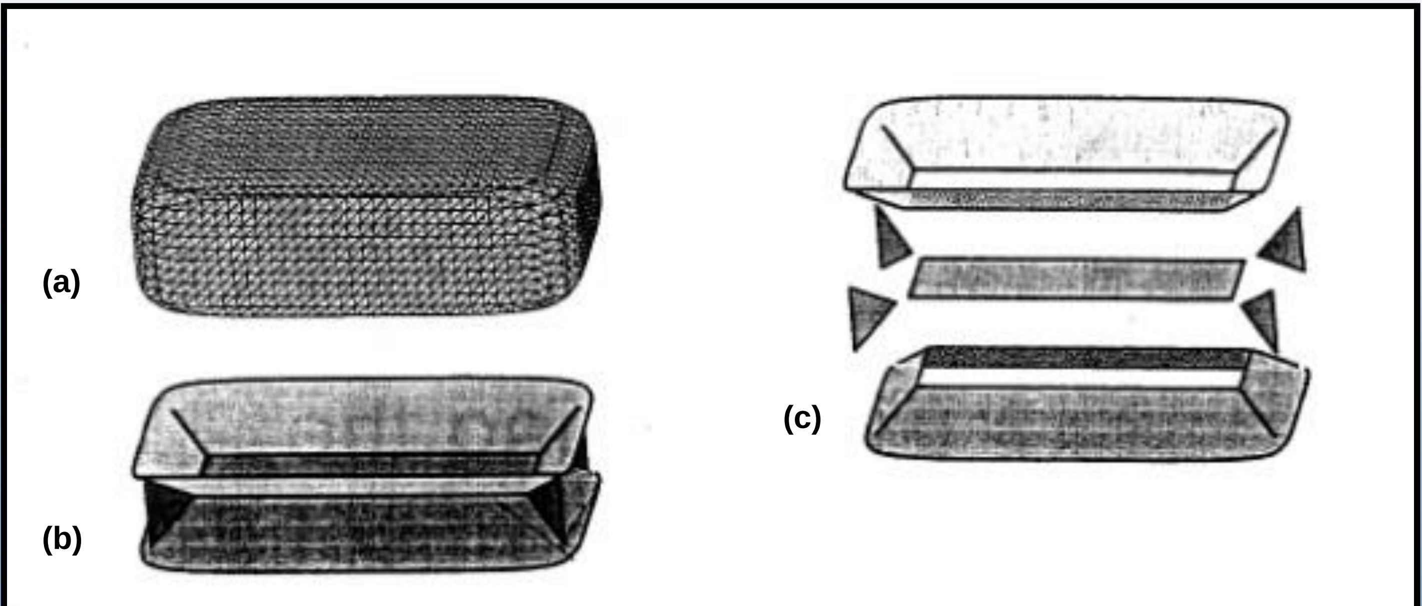
The medial surface of a 3D solid is the closure of the locus of centers of maximal inscribed spheres.




Finite Element Modelling Group, Queen's University, Belfast

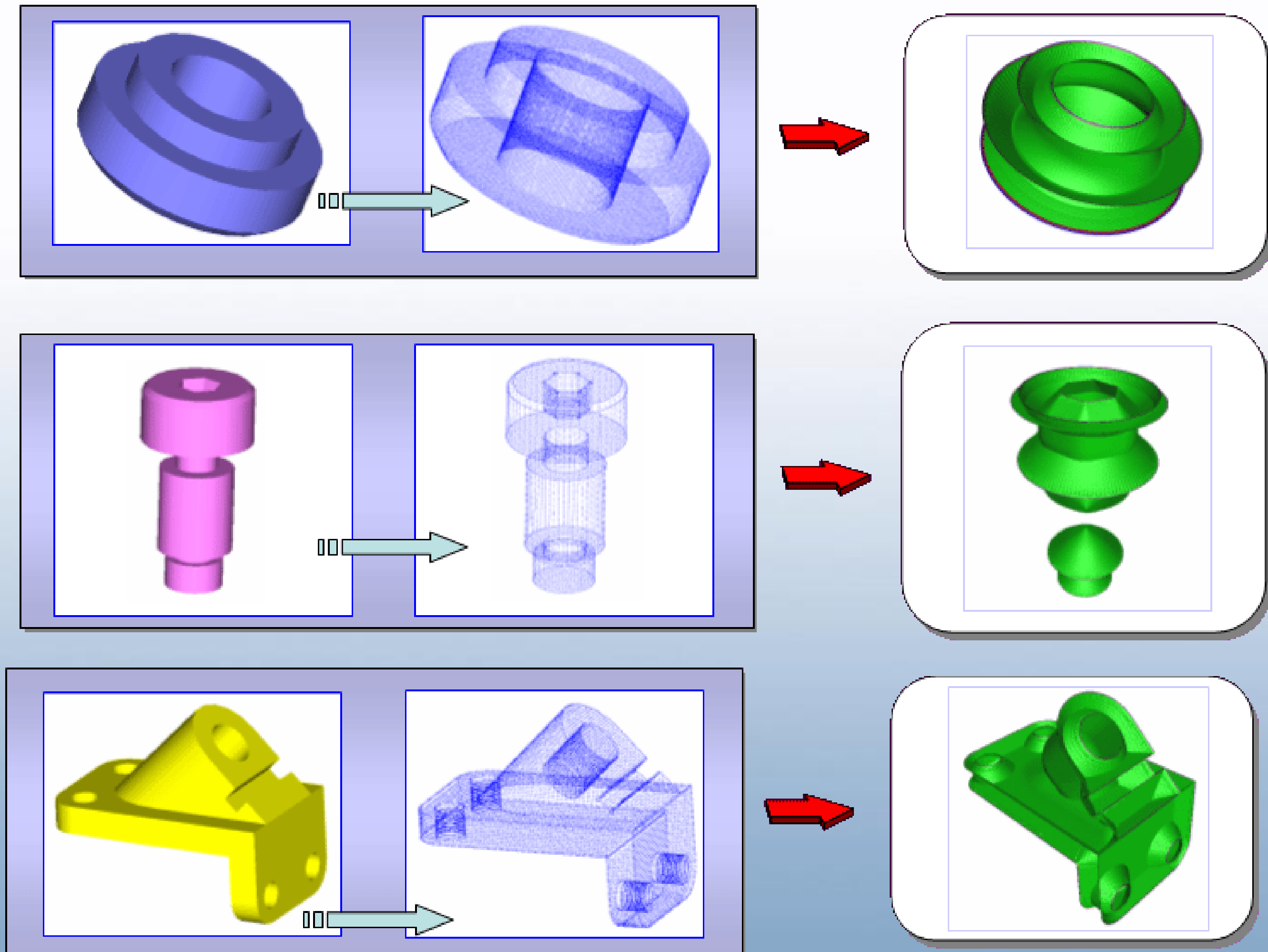
# Morphology of 3D skeleton

(a) super-ellipsoidal block;
(b) skeleton of block with various element labeled;
(c) exploded skeleton showing separated patches.




Images by G.Turkiyyah et al.

# Medial axis representations



Properties:

- Unambiguous
- Unique
- Efficient

in meshing algorithms

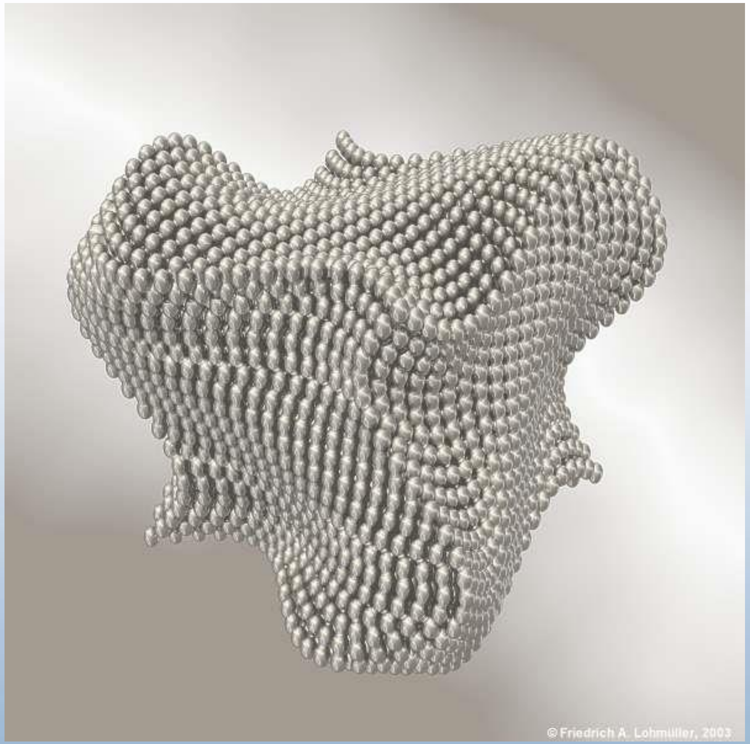
Images by T. K. Dey, H. Woo and W. Zhao

# Contents

- Ray representations
- Parametric function representations
- Real function representations
- Hybrid schemes and conversions

3. Solid modelling systems

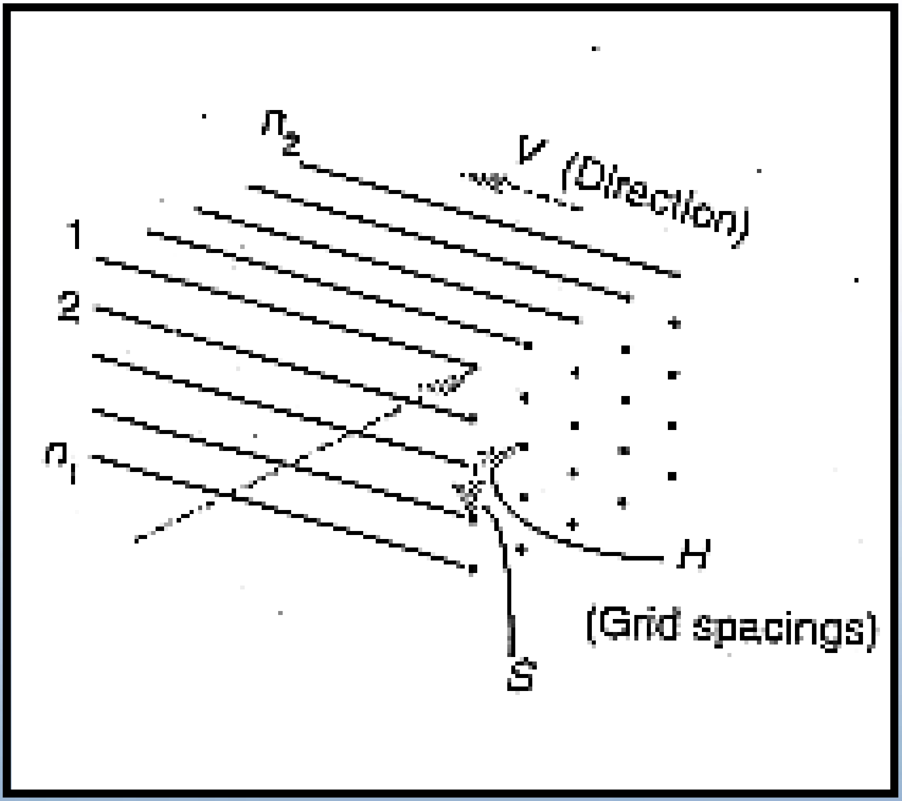
4. References




*"Dancing Cube" by Friedrich A. Lohmueller (2003)*

# Ray representations

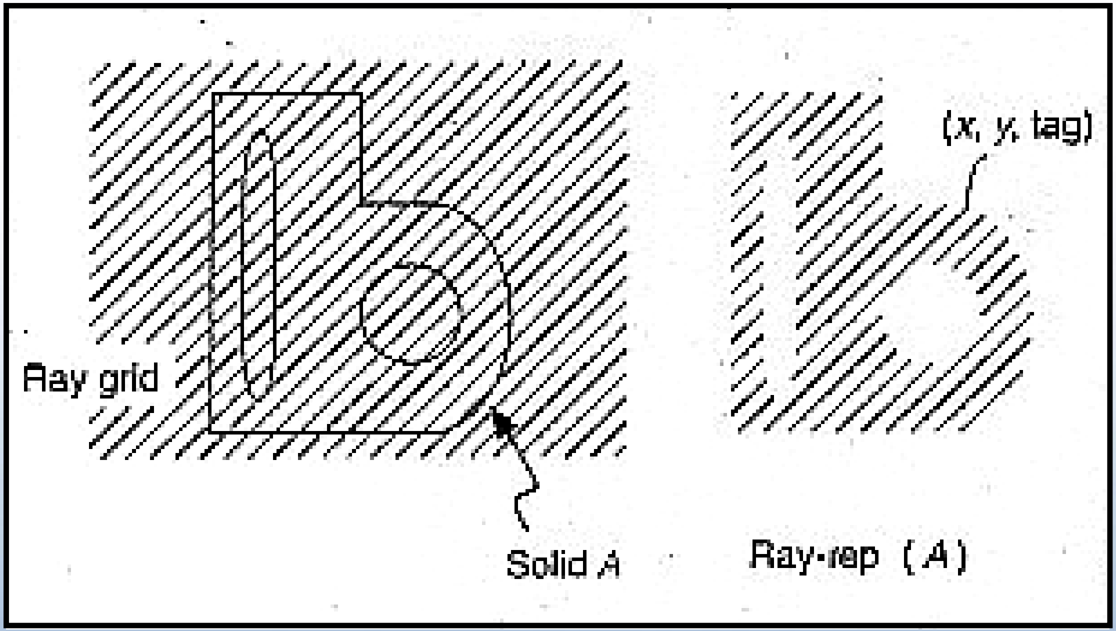
A ray grid is a finite set of regularly spaced parallel lines with an associated direction V.




J. Menon et al. CG&A, vol.14, N3, 1994

A ray representation of a solid is a set of segments resulting from the intersection of the ray grid with the solid.

A ray-rep might also contain tags,




descriptive symbolic information appended to the segments. Tags can identify the primitive half-spaces in solid's CSG or faces in its B-rep.

Properties:

- Unambiguous

  under suitable conditions and with appropriate tags. Can be converted exactly from and to CSG and B-rep

- Not unique

  Depends on grid spacing

# Parametric function representations

Shapes are represented by multidimensional, continuous, piecewise-differentiable parametric functions:

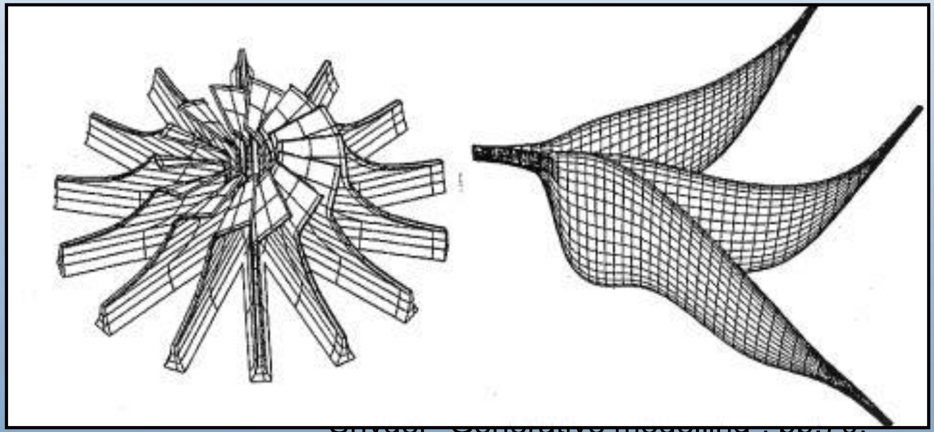
$$F: R^n \rightarrow R^m$$

where $R^n$ is parameter space and $R^m$ is object space.

For n=2, m=3

[x(u,v), y(u,v), z(u,v)]

defines a surface in 3D space.



Snyder "Generative modelling", pp.76,

95

# Properties

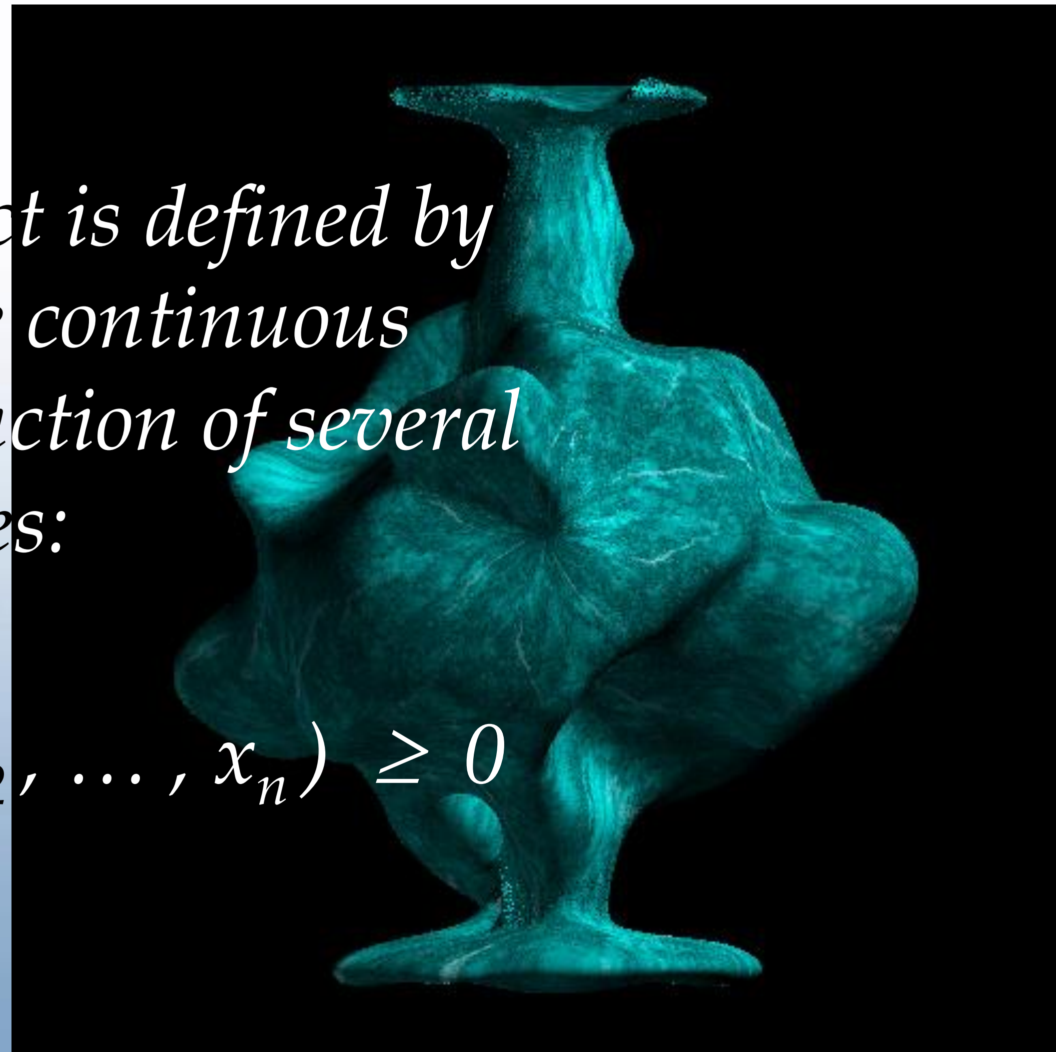
- Extension and generalization of sweeping
- Unambiguous, non unique representation
- Compact and easy to create
- Efficient algorithms

  with the use of interval analysis

# *Real function representations*

*An object is defined by a single continuous real function of several variables:*

$$f(x_1, x_2, \dots, x_n) \geq 0$$



- A function *f* can be defined analytically, with an evaluation algorithm, or with sampled values and an appropriate interpolation procedure.

# Function representation FRep

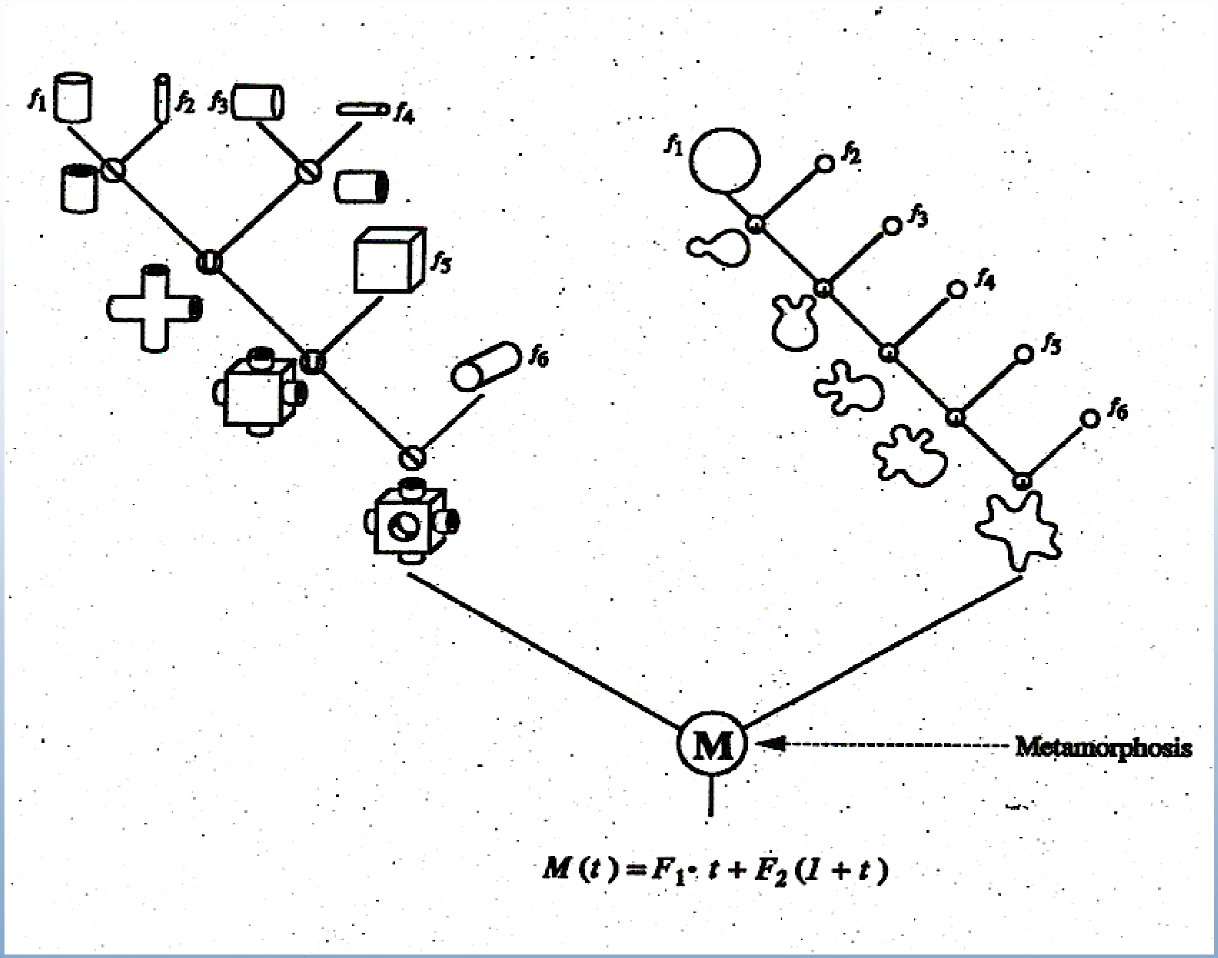
- Uniform representation of multidimensional objects defined as

$$F(X) \geq 0$$

- Function $F(X)$ evaluation procedure traversing the construction tree structure
- Leaves: primitives
- Nodes: operations + relations
- System extensibility

FRep

# Construction of metamorphosis



# Properties

- Closed under the arithmetic, set-theoretic, Cartesian product, projection and other operations.
- The abstraction level is higher than that of other known representations.

  Combinations of the following modelling styles are supported: CSG, sweeping, implicit and volumetric objects.

# Real function representations
# Properties

- Unambiguous

  A function defines one object

- Not unique

  for example, $k \cdot f \geq 0$

- Concise and easy to create

- Efficient

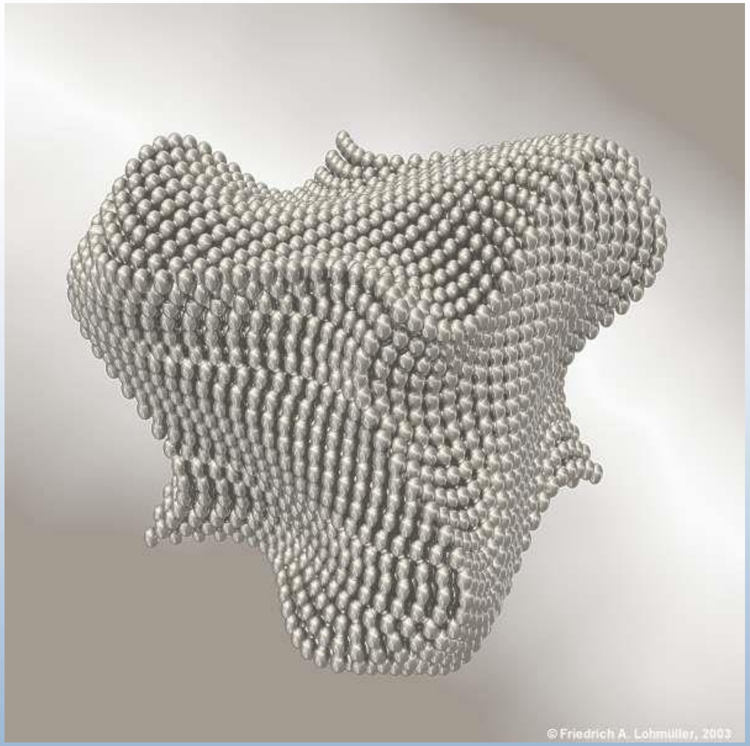
  in modelling highly complex objects

# Contents

- Ray representations
- Parametric function representations
- Real function representations
- Hybrid schemes and conversions

3. Solid modelling systems

4. References




*"Dancing Cube" by Friedrich A. Lohmueller (2003)*

# Hybrid schemes and conversions

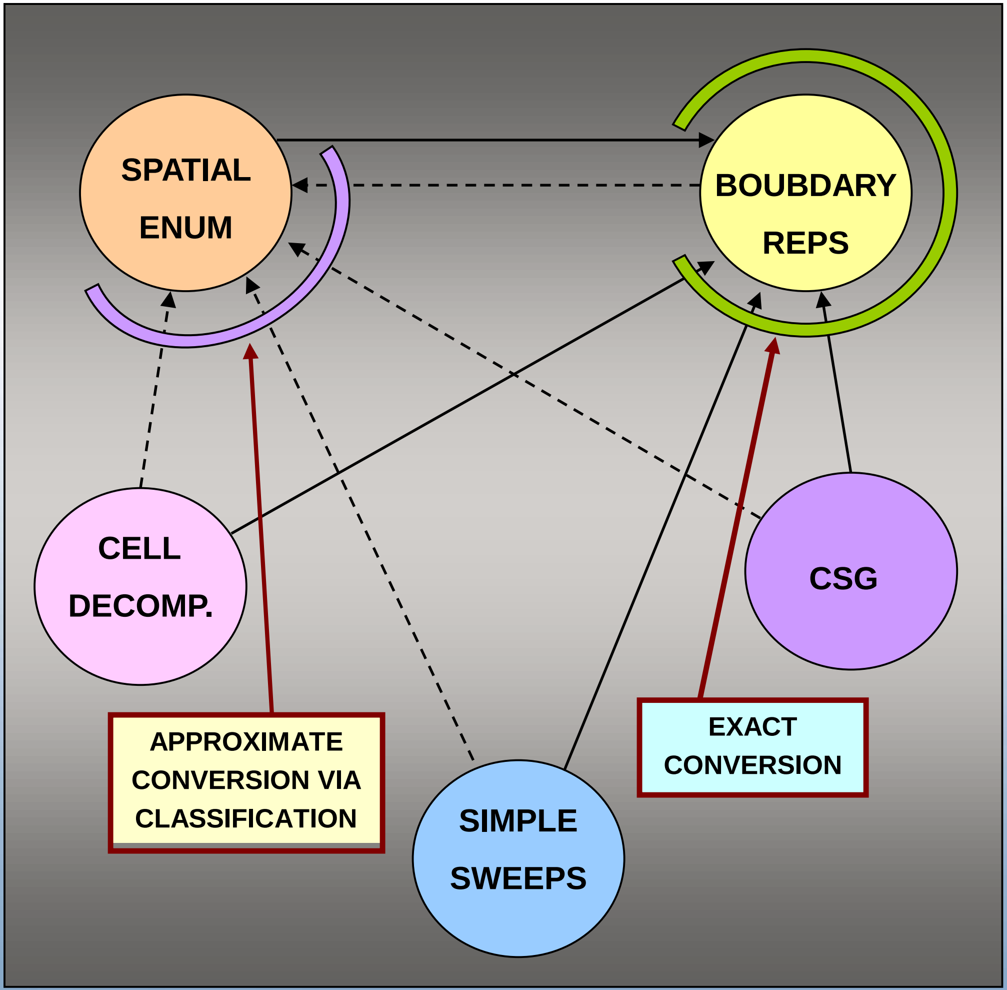
- ***Hybrid***, or ***non-homogeneous***, representation schemes may be designed by combining several schemes:

- 1) CSG/ boundary hybrid: CSG-like trees whose leaves are primitive solids or B-rep non-primitive solids.

  Is used as the basis for the input language of some systems.

# Hybrid schemes and conversions

- 2) CSG/sweep hybrid: CSG-like trees whose leaves may be solid-sweep representations. Useful in numerically controlled (NC) machining and computer vision.




Requicha, Comp. Surveys, p.456

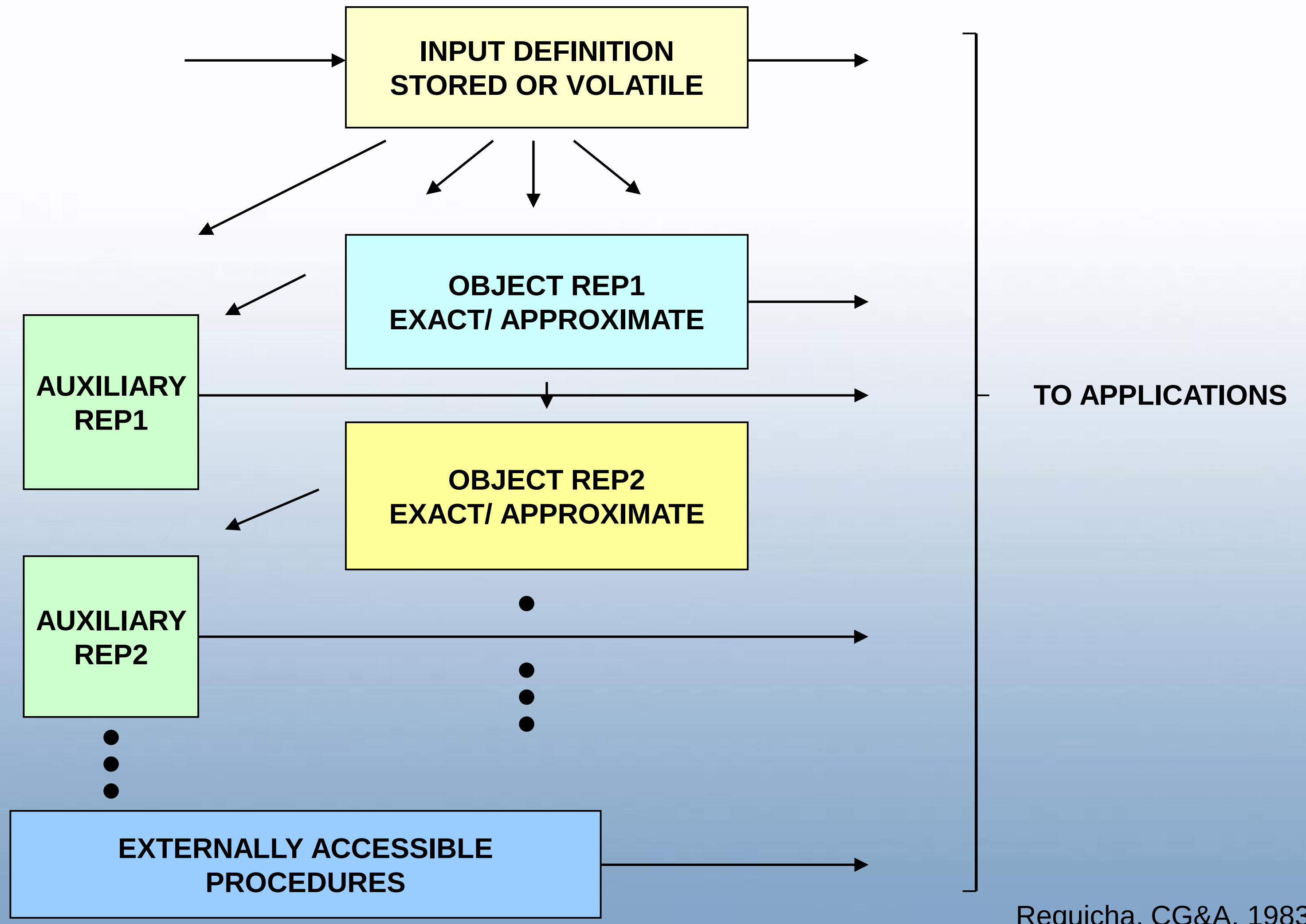
- Reasons for the lack of bidirectional exact conversions:

1) schemes such as sweeps have smaller domain than CSG, B-rep, or cell decompositions;

2) algorithms are not known.

- Exact conversion from CSG to B-rep ("*boundary evaluation*") requires nontrivial algorithms.

# Architectures of solid modelling systems

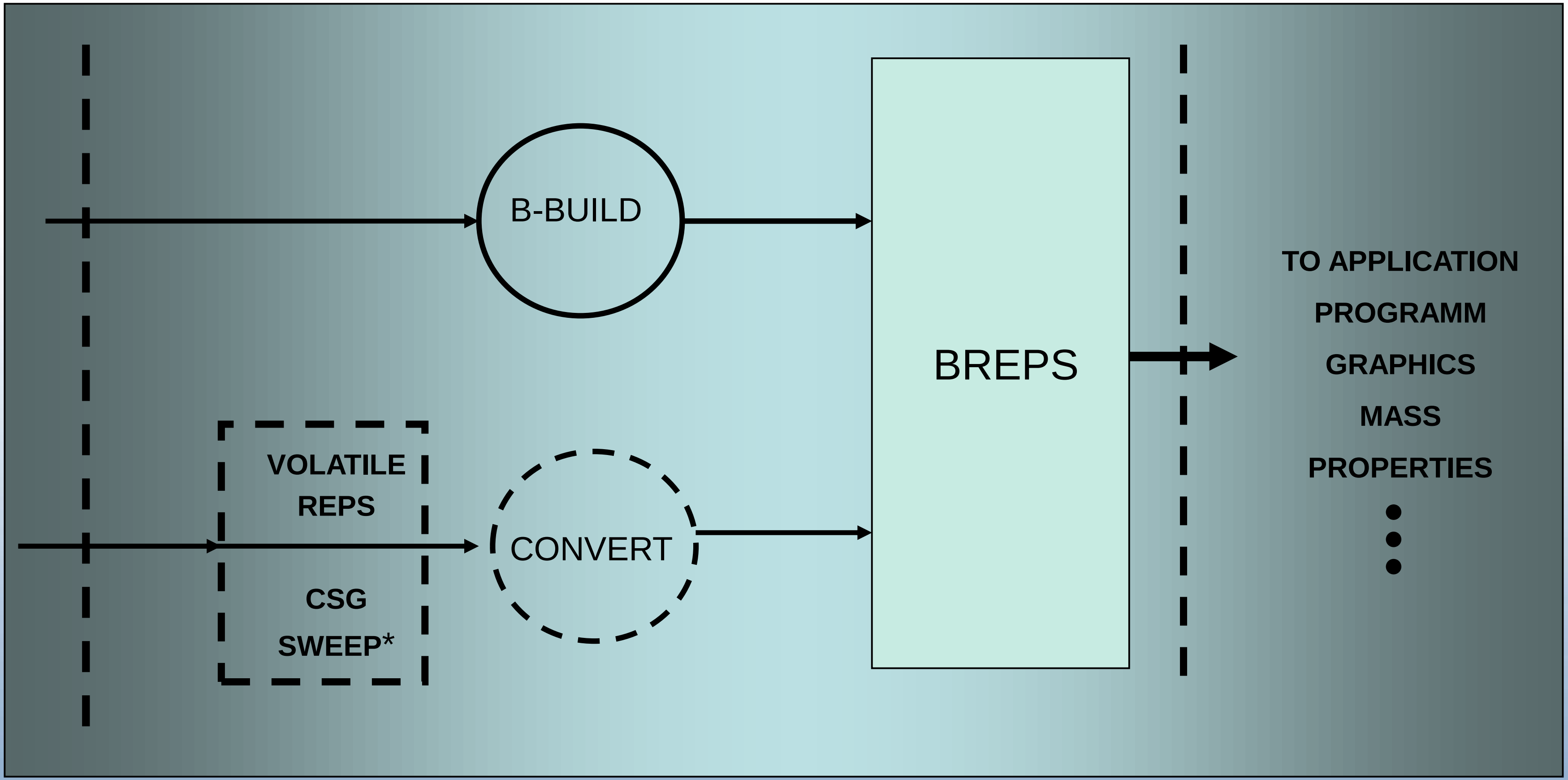



Requicha, CG&A, 1983, p.28

# Architectures of solid modelling system

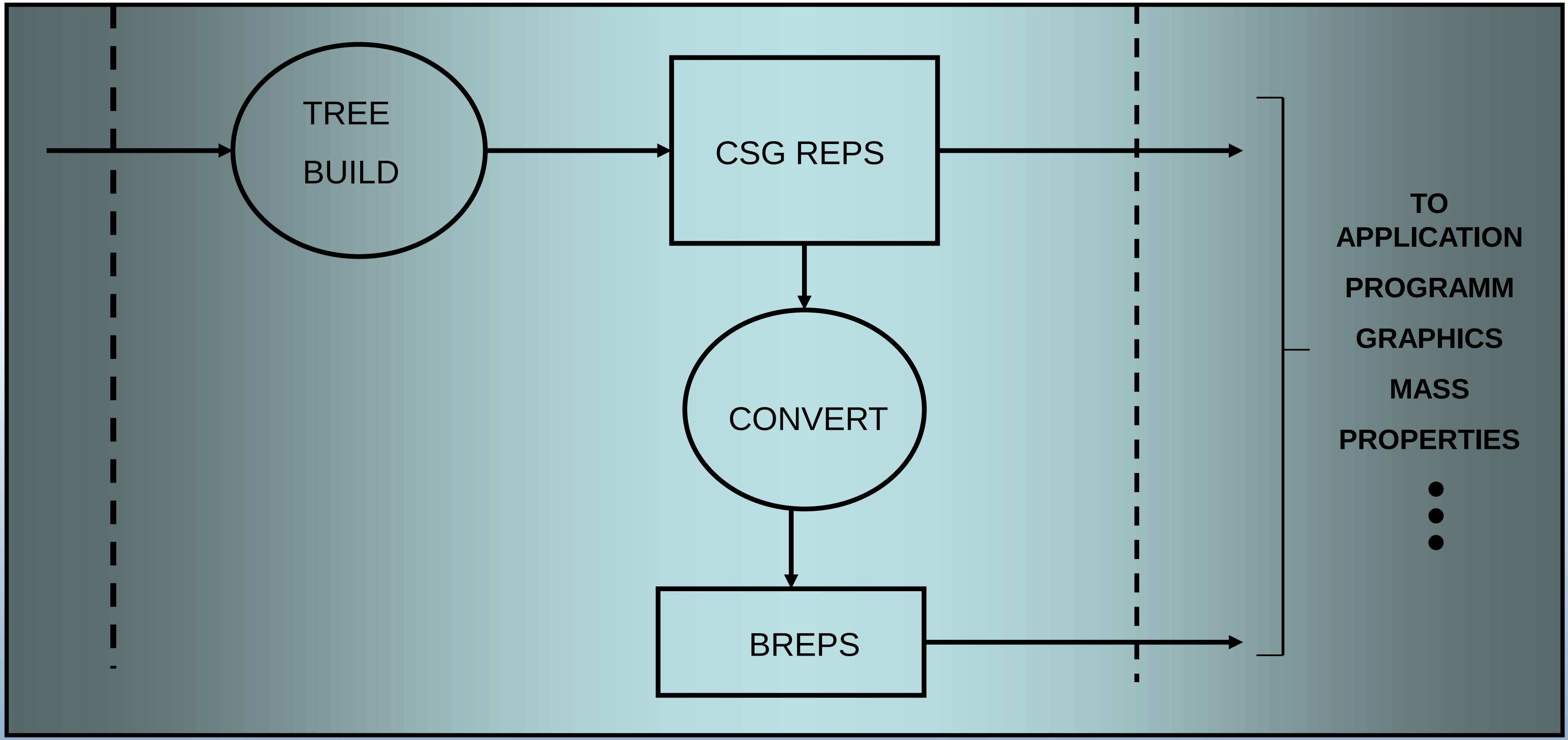
- A solid modelling system is regarded as
- 1) a specific collection of representations (exact, approximate, auxiliary) with at least one being valid and complete;
- 2) a collection of procedures for managing representations, conversions, and other geometrical calculations.
- Applications: graphics, mass properties calculations, finite-element meshing, interference checking and path planning, mechanism simulation, rapid prototyping, manufacturing, data storage and exchange.

# Single representation and hybrid systems




Single (boundary) representation systems

# Single representation and hybrid systems




Hybrid CSG/ BRep systems

**Representational facilities of selected geometric modeling systems.**

| SYSTEM | PRIMARY REP SCHEME | DOMAIN (DEFINED BY HALFSPACES) | OTHER CONSISTENT REPS | INPUT BASED ON | INPUT MODALITIES |
|---|---|---|---|---|---|
| SHAPES | CSG: HALFSPACES | QUAD | — | CSG | SC;T/I;IG* |
| TIPS | | "ARBITRARY" | — | CSG | T/B |
| GDP/GRIN | CSG: BOUNDED PRIMITIVES | QUAD | ≃B-REP | CSG | SC;IG |
| PADL-1 | | PL & CYL (⊥) | B-REP | CSG | T/I |
| PADL-2 | | QUAD | B-REP | CSG | SC;T/I |
| SYNTHAVISION | | QUAD, SS | WIREFRAME | CSG | T/B |
| GMSOLID | HYBRID CSG/B-REP | QUAD | B-REP | CSG, SWEEP* | IG |
| U.M./BORKIN | B-REP | PL | — | CSG, SWEEP* | T/I;IG* |
| BUILD-2 | B-REP | QUAD | — | CSG, EOP | SC;T/I |
| CADD | B-REP | QUAD, SS | — | SWEEP* | IG |
| COMPAC | B-REP | QUAD | — | CSG, SWEEP* | T/B? |
| DESIGN | B-REP | QUAD | — | CSG | IG |
| EUCLID | B-REP | ≃QUAD | — | CSG, SWEEP* | T/I;IG? |
| GLIDE | B-REP | PL | — | CSG, SWEEP*, EOP | SC;T/I |
| MEDUSA | B-REP | ≃QUAD, SS | — | CSG, SWEEP* | IG |
| PROREN-2 | B-REP | QUAD | — | CSG, SWEEP* | T/I |
| ROMULUS | B-REP | QUAD | — | CSG, SWEEP*, EOP | SC;T/I |

PL = PLANE
QUAD = QUADRIC SURFACES (NOTE 1)
SS = SCULPTURED SURFACES
⊥ = ORTHOGONAL POSITIONING
≃ = APPROXIMATE
SWEEP* = TRANSLATIONAL AND/OR ROTATIONAL SWEEP

EOP = EULER OPERATIONS
SC = SUBROUTINE CALL
T/I = TEXT/INTERACTIVE ORIENTATION
T/B = TEXT/BATCH ORIENTATION
IG = SOPHISTICATED INTERACTIVE GRAPHICS
IG* = SIMPLE INTERACTIVE GRAPHICS

Requicha and Voelcker 1982

# More late and modern systems

## Kernel modelers

- Parasolid (EDS Unigraphics, Cambridge, UK)
  BRep solid modeler supporting free-form surfaces (development of Romulus)
  **http://www.eds.com/**
- ACIS (Spatial Technology, USA)
  B-rep object-oriented toolkit (development of BUILD)
  **http://www.spatial.com/**
- Designbase (Ricoh, Japan)
  B-rep solid modelling library with free-form surfaces
  www.ricoh.co.jp/designbase/
- SVLIS (Information Geometers, UK)
  CSG object-oriented kernel modeller
  **http://www.bath.ac.uk/~ensab/G_mod/Svlis/svlis.html**

# B-rep modelers

- CATIA **(Dassault, France / IBM)**
  **http://www.catia.ibm.com/**

- AutoCAD **Release (Autodesk, USA)**
  **based on ACIS**
  **http://www.autodesk.com/**

- SolidWorks **(SolidWorks, USA)**
  **PC+Windows interactive system based on Parasolid**
  **http://www.solidworks.com/**

# CSG based hybrid systems

- Ray Casting Engine RCE (Duke/ Cornell Universities, USA) VLSI parallel special-purpose computer based on CSG/ ray-rep

  http://www.cs.duke.edu/~kedem/RCE/RCE.html

- POVRay CSG/B-rep hybrid modeller for photorealistic rendering
  http://www.povray.org

- BRL-CAD (USA Army) CSG/ BRep solid modelling system

  http://www.brl-cad.org

# More late and modern systems

## FRep based systems

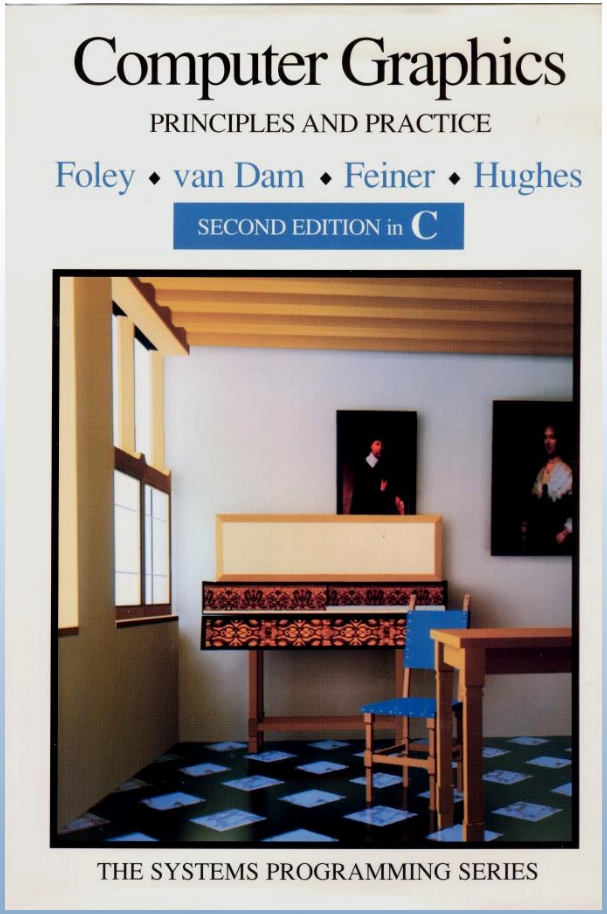
- HyperFun (Aizu-Hosei, Japan; Bournemouth, UK) special-purpose high-level language and tools

  http://www.hyperfun.org

- Symvol for Rhino (Uformia, Norway) FRep plug-in to Rhinocerous CAD system http://uformia.com/products/symvol-for-rhino/

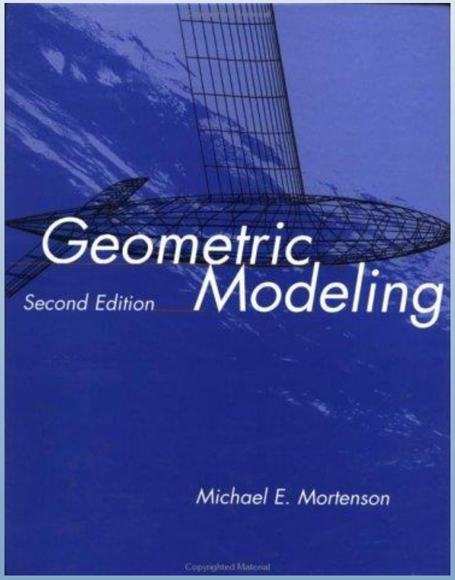
# References

- James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics: Principles and Practice (2nd Edition in C ), Addison-Wesley, Reading, MA, 1997.



# References


- Aristides A. G. Requicha “Representations of rigid solids: theory, methods, and systems”, computing Surveys, vol. 12, No. 4, 1980, pp. 437-465.
- A. A. G. Requicha, H. B. Voelcker “Solid modelling: a historical summary and contemporary assessment”, IEEE Computer Graphics and Applications, vol.2, No.2, 1982, pp. 9-24.
- Michael E. Mortenson, Geometric modelling, John Wiley and Sons, 1985. Second edition, 1997. Third edition, 2006.

# References


- Christoph M. Hoffmann “Geometric and Solid modelling. An Introduction”,Morgan Kaufmann Publishers, 1989, 338 p.
- A.A.G. Requicha, J.R. Rossignac “Solid modelling and beyond”, IEEE Computer Graphics and Applications, vol.12, September, 1992, pp.31-44.
- Snyder, John M., Generative modelling for Computer Graphics and CAD: Symbolic Shape Design Using Interval Analysis, Academic Press, 1992.

# References


- Menon J., Marisa RJ, Zagajac J., More powerful solid modelling through ray representations, IEEE Computer Graphics and Applications, vol.14, No.3, 1994, pp.22-35.

- J. Rossignac. Through the cracks of the solid modelling milestone, From Geometric modelling to Advanced Visual Communications, S.Coquillart, W. Strasser, P. Stucki, Springer Verlag, pages 1-75, 1994.

# References

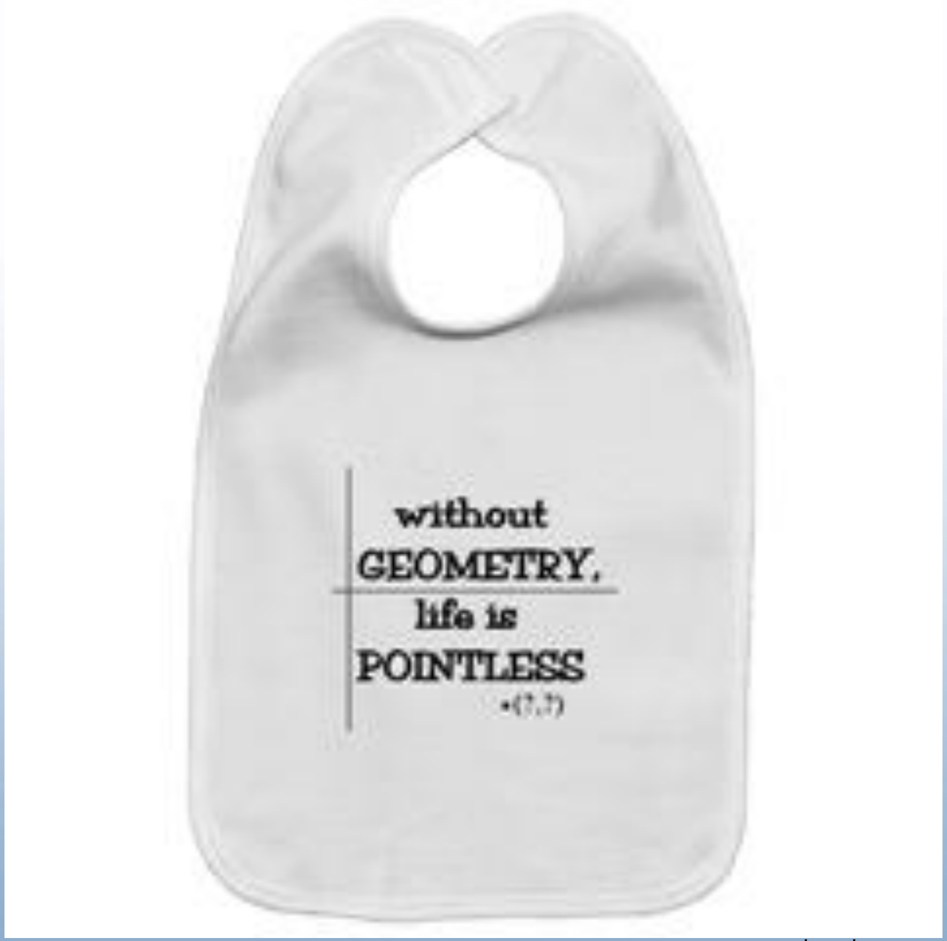

- Herbert Voelcker "Support for solid freeform fabrication", NSF Workshop on Design Methodologies for Solid Freeform Fabrication, 1995.
- Christoph M. Hoffmann, Jarek Rossignac: A Road Map To Solid modelling, IEEE Trans. Vis. Computer Graphics, Vol.2, No.1, pp. 3-10,1996.
- A.A.G. Requicha "Geometric modelling: A First course", 1999.
- Martin Culpepper, Understanding how things work, Lecture course, MIT,1999.

# Words of wisdom

*"...we are geometricians only by chance"*

*Dr. Johnson*

*"Without geometry, life is pointless"*




www.proudnerd.com