

Requirements to the Scene Data Base

Andrei B. Khodulev

Keldysh Institute of Applied Mathematics RAS

Edward A. Kopylov

Moscow State University

Dmitry D. Zdanov

Vavilov State Optical Institute, St.Petersburg, Russia

International Conference Graphicon 1998,
Moscow, Russia, <http://www.graphicon.ru/>

Interfaces for Ray Tracing

- RenderMan
- Mental Ray
- RenderDrive

- **AR250** (Advanced Rendering Technology Ltd.)
- Using a highly pipelined parallel architecture the AR250 performs 4000 million microprocessor equivalent floating point instructions per second. This focused computational power will allow it to perform up to 80,000,000 intersection tests per second, roughly 15 times the performance of a graphics workstation.

What is Scene Data Base ?

- The scene database is an in-memory representation of 3D objects used by program
- +
- attribute support
- read / save
- import / export

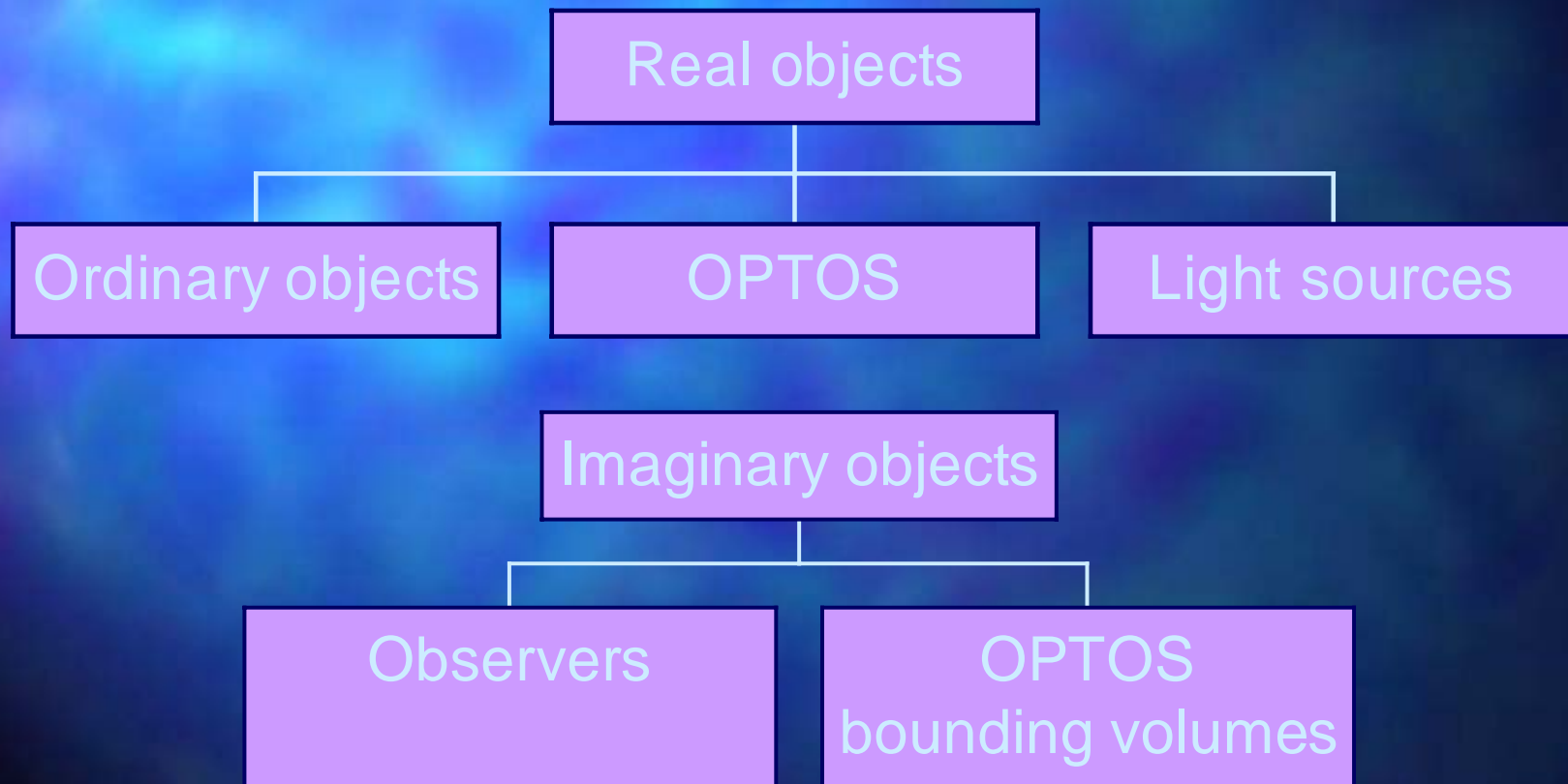
What is scene ?

- Scene is a model of some part of real world
- Scene consists of objects
- Each object is characterized with its shape (geometry) and physical properties

Operations with scenes

- scene creation/modification
- optical simulation in the scene
- visualization of the scene and simulation results
- ---
- The SDB keeps all info about scene
- The SDB returns some info about scene
- The SDB makes changes in scene

Object classes



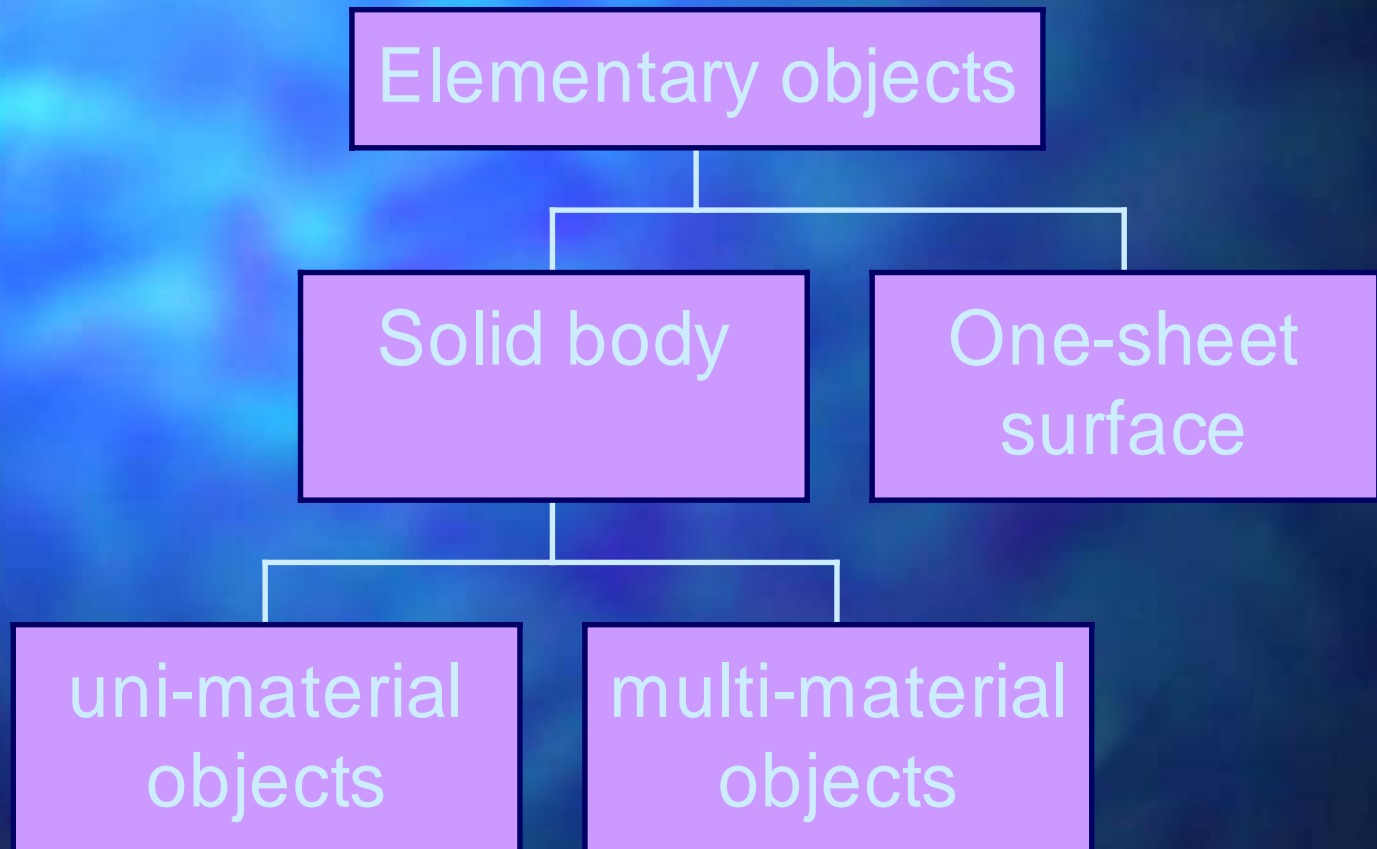
object-oriented approach

- Open architecture
- Object hierarchy
- Encapsulation

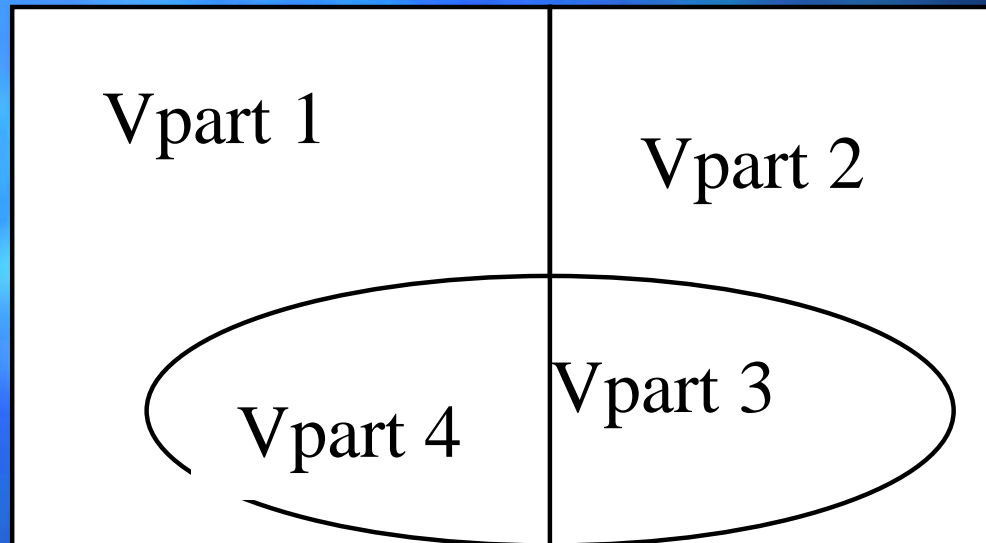
The set of allowed operations

- extraction of object geometry in form of triangular mesh
- geometric transformation (rotation, translation, etc.) of an object
- ray tracing through the scene (RTM)

Object concept



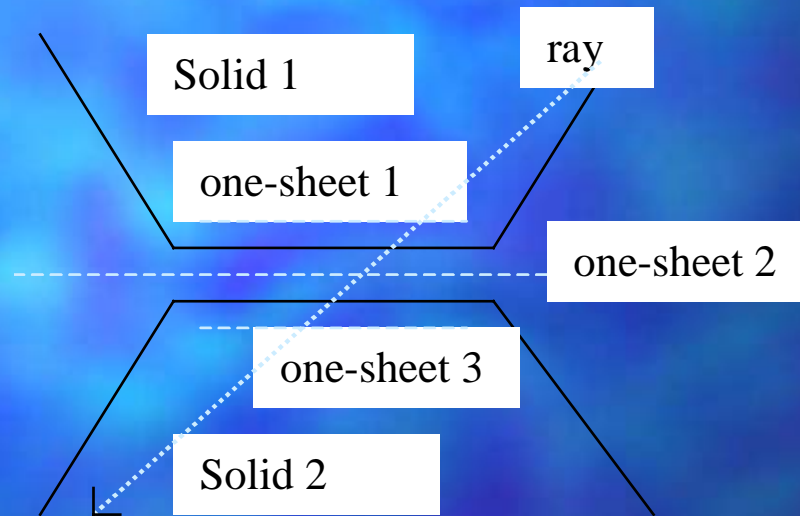
Multi-material objects



- Volume parts (Vparts)
- Surface parts (parts)

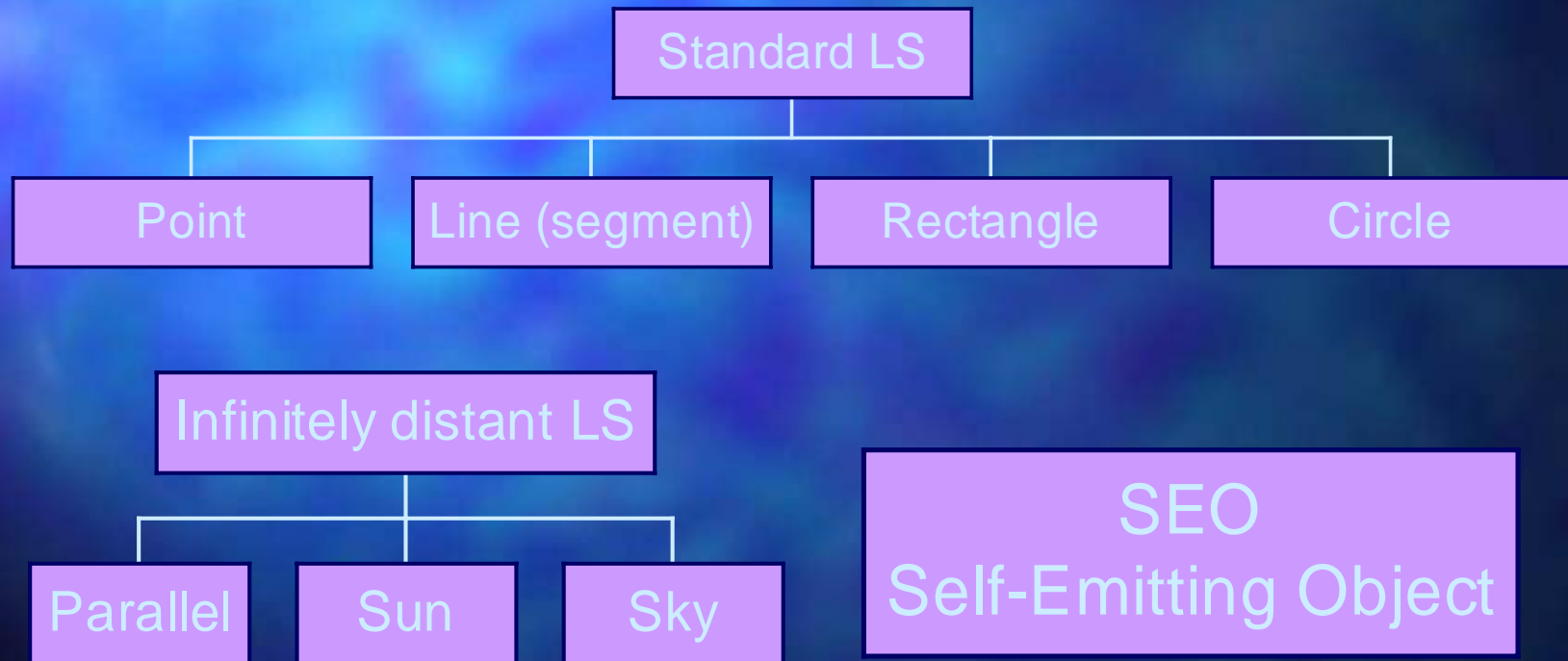
- Curved surfaces
- Constructive Solid Geometry (CSG)
- Volumetric Boolean operations (VBO)
- (union, intersection and difference)
- VBO for multi-volume solids
- Forms of “Union”: disjoint, glue, merge
- Procedural shapes/attributes
- Surface orientation (SEO, BRDF)
- Coplanar surfaces
- Light sources

Most complex allowed case of surface coplanarity



- 5 boundaries: 2 belonging to touching solids and 3 sheet objects lying in Solid 1, Solid 2 and between them

Light sources



Special Objects

- inform about the fact of ray intersection: clipping planes, bounding boxes
- accumulate some ray info in point of ray intersection: observers
- ---
- SDB supports placement of observers and assignment of their parameters

Observer types

- Camera
- Section sector
- Plane
- Goniometric
- Volume

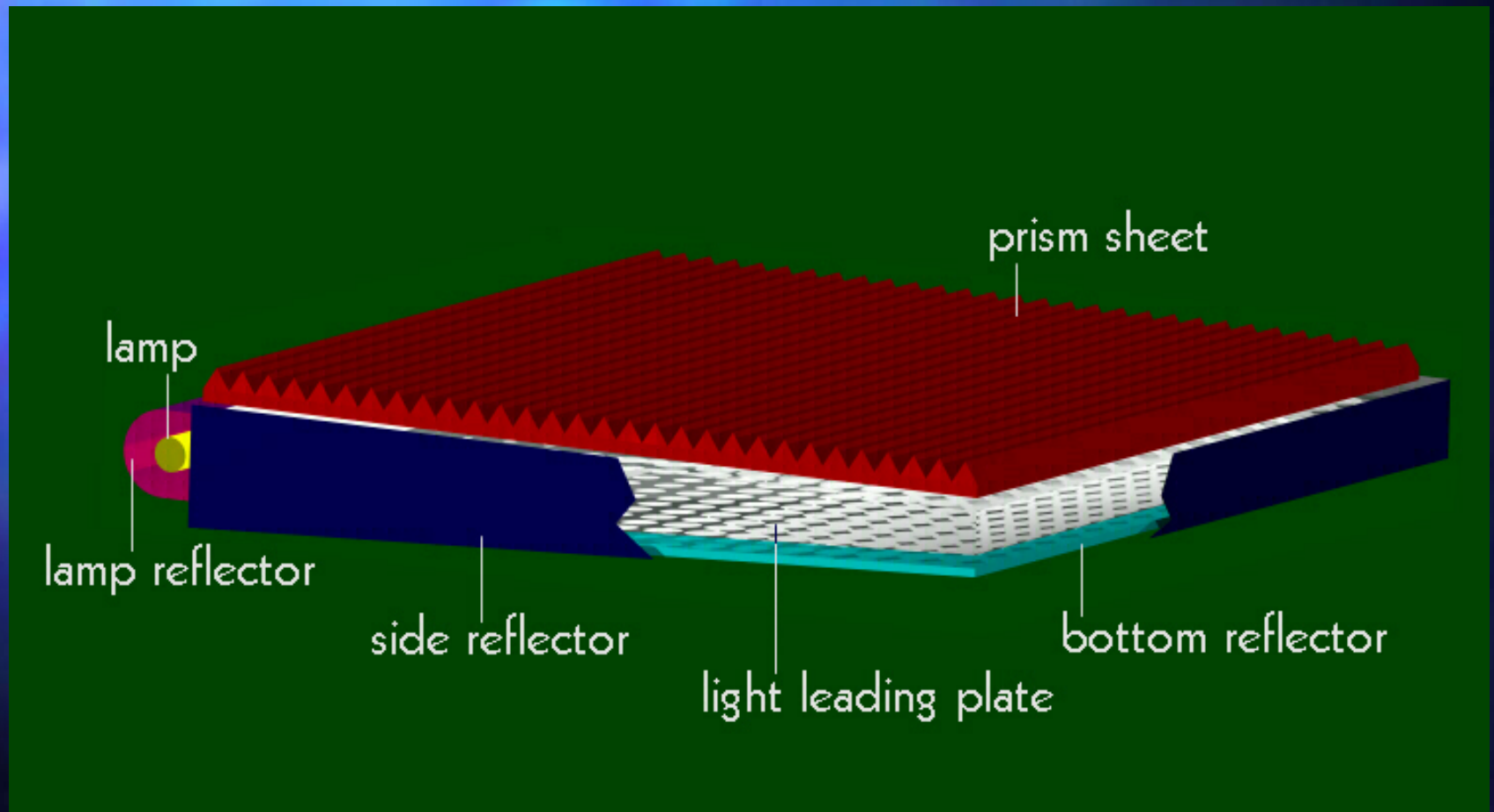
Attributes

- assign/change/request attributes for any vpart, any surface and any special object
- Texture assignment
- Orientational textures
- Spatial dependency of attributes: procedural 2D textures, procedural BRDF and luminaire distribution
- Multi-product support

OPTICAL ELEMENTS (OPTOS)

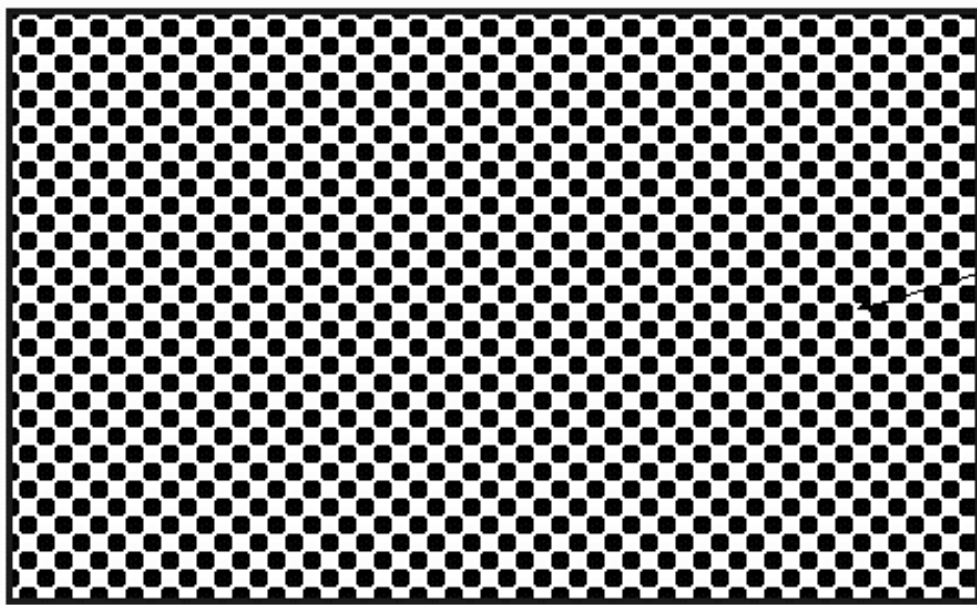
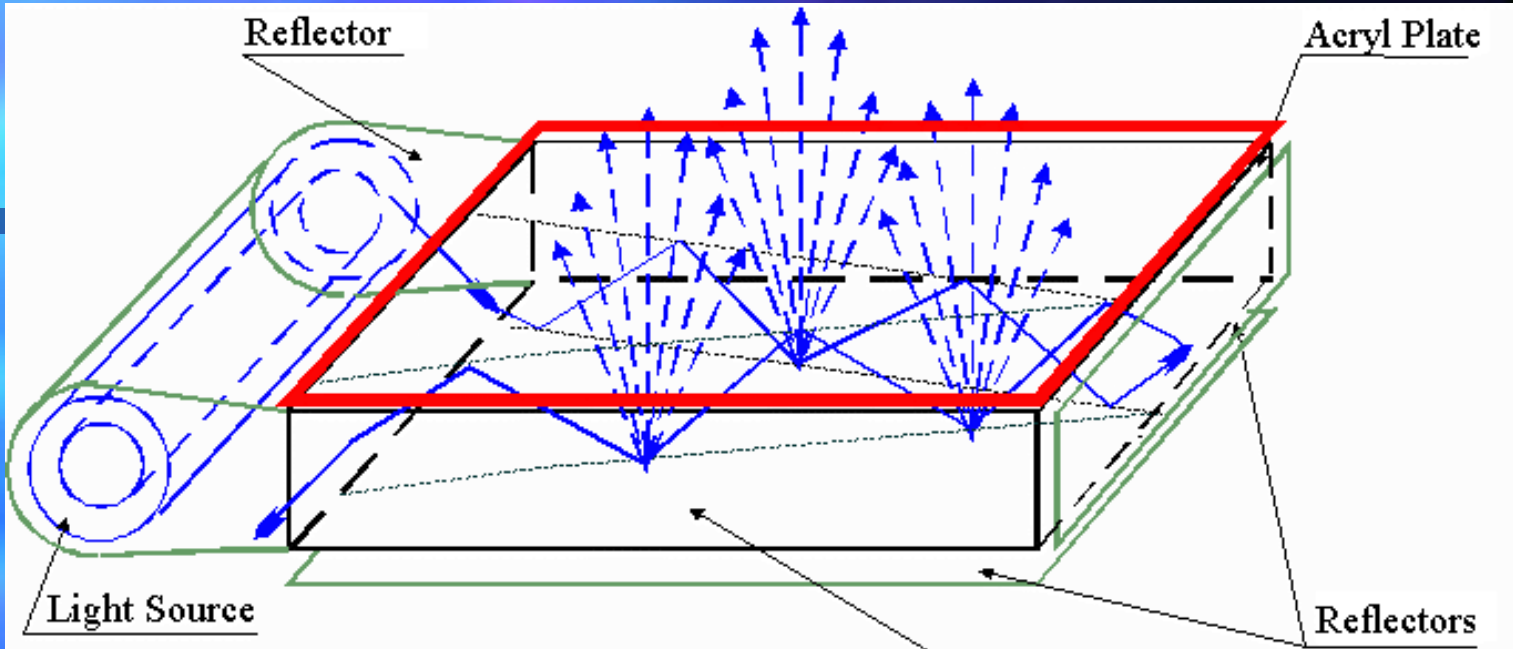
- plate prisms
- Fresnel's lenses
- Fiberglass lens arrays (FLA)

Plane Light Emitter Device



International Conference Graphicon 1998,
Moscow, Russia, <http://www.graphicon.ru/>

Plane Light Emitter Device



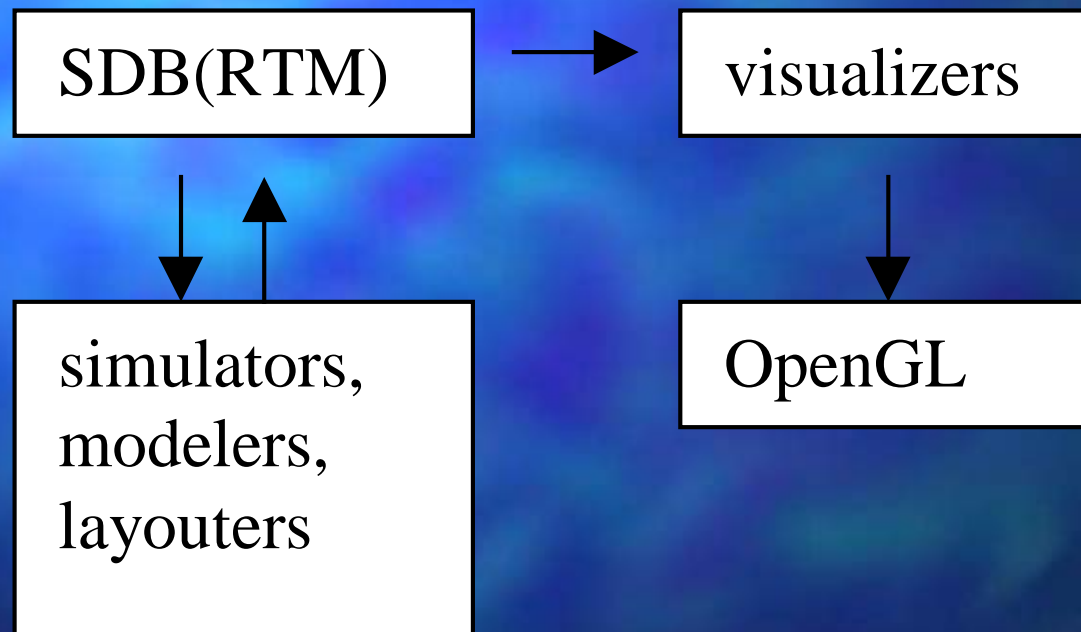
Imprinting Diffuse Plate

DISTRIBUTED SDB

- share the same scene between several callers located at several computers connected via network

SDB SERVICES

- Visualizer service



Visualizer services

- Object bounding box
- Explicit form generation: triangle mesh with representation accuracy control
- Projection to/from explicit form

Scene modification services

- Hierarchy representation: combining a compound object from several other ones; destroying hierarchy; application of geometric transformations; etc.
- Possible transformations: translation in space; rotation; scaling factor; etc.
- Multiple copy
- Dynamic changes efficiency
- Data Integrity

Additional SDB services

- Correctness check: collision detection; closeness detection; vpart encapsulation detection
- Incremental correctness check
- Optimization check
- Scene input/output, import/export

EFFICIENCY CONSIDERATIONS

- ray tracing
- projection internal form to explicit form