

The Fun of Connection

Abstraction for Data Systems

Chet Mancini

Github & Twitter: @chetmancini

LEGO **BASIC**

545





Duplo®

Godtfred K. Christiansen
US Pat. 3597105 / Aug. 1971

Fischertechnik®
K'Nex®

Artur Fischer
US Pat. 3464147 / Sep. 19, 1966

Krinkles®
Lego®

Joel Glickman & M. Doesner
US Pat. 5061219 / Dec. 11, 1990

Walter Heubl
US Pat. 3603025 / Sep. 30, 1968

Lincoln Logs®
Tinkertoy®
ZomeTool®
Zoob®

Godtfred K. Christiansen
US Pat. 3005282 / Jul. 28, 1958

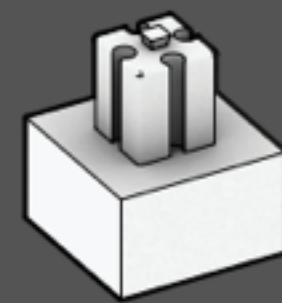
John Lloyd Wright
US Pat. 1351086 / Jan. 8, 1920

Charles H. Pojeau
US Pat. 1113371 / Jul. 8, 1914

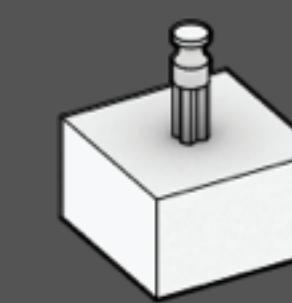
Steven Rogers & P. Hildebrandt
US Pat. 6840599 / Nov. 1, 2002*

Michael J. Grey
US Pat. 5097417 / Dec. 11, 1990*

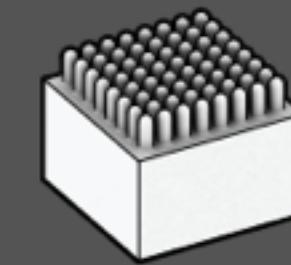
(Duplo to Duplo)



uck-00f01m

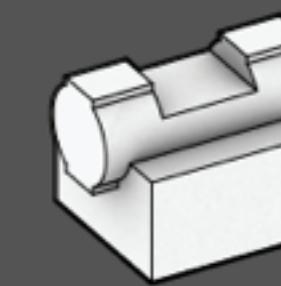


uck-00f03m

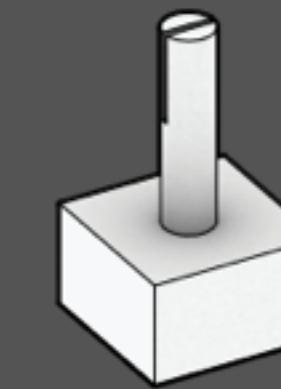


uck-00f04m

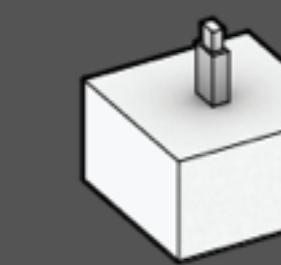
(Lego to Duplo
already supported
by manufacturer)



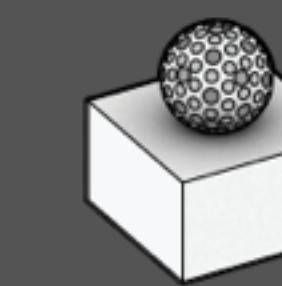
uck-00f06m



uck-00f07m

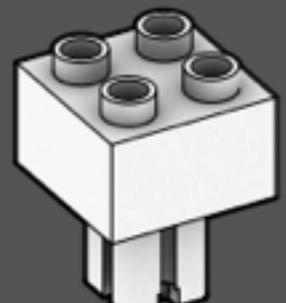


uck-00f08m



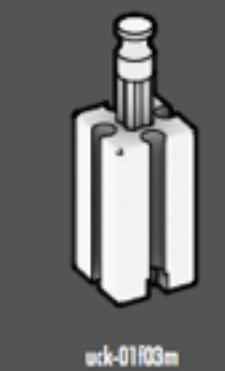
uck-00f09m

Duplo®



uck-01f00m

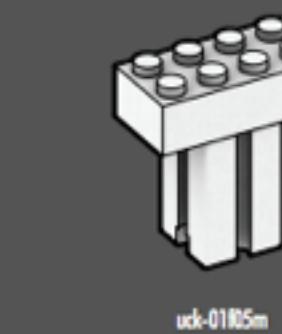
(Fischertechnik to
Fischertechnik)



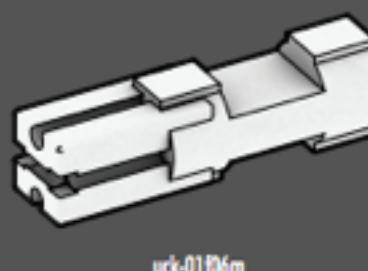
uck-01f03m



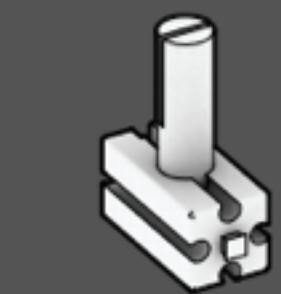
uck-01f04m



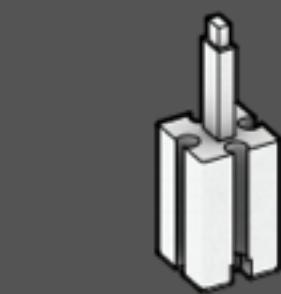
uck-01f05m



uck-01f06m



uck-01f07m

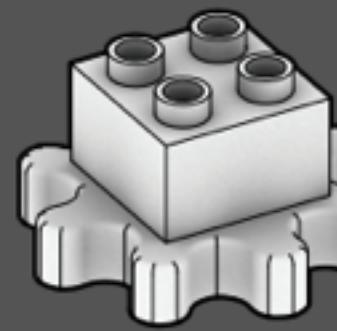


uck-01f08m

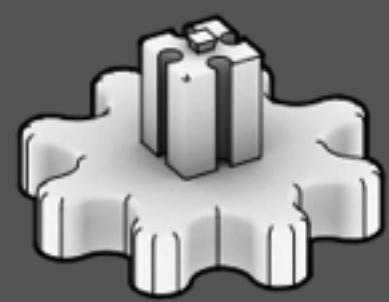


uck-01f09m

Fischertechnik®



uck-02f00m



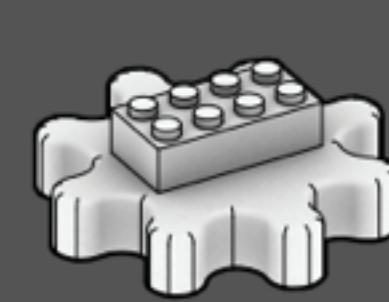
uck-02f01m



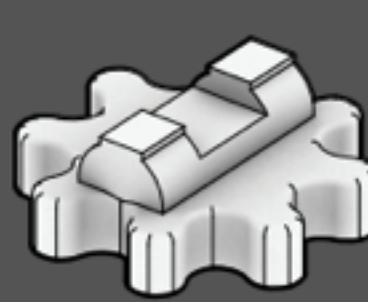
uck-02f03m



uck-02f04m



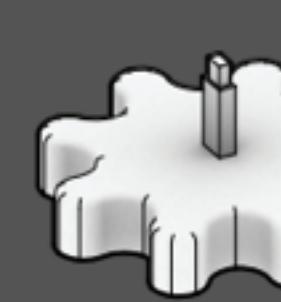
uck-02f05m



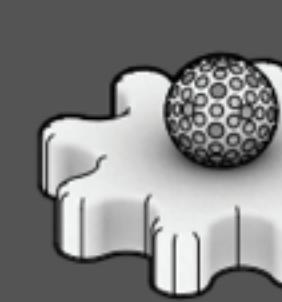
uck-02f06m



uck-02f07m



uck-02f08m



uck-02f09m

Gears!
Gears!
Gears!



uck-03f00m

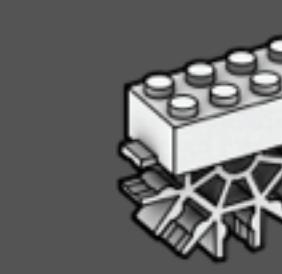


uck-03f01m

(K'Nex to K'Nex)



uck-03f04m



uck-03f05m



uck-03f06m



uck-03f07m



uck-03f08m



uck-03f09m

K'Nex®



(K'Nex to Krinkles)



uck-04f00m



uck-04f01m



uck-04f02m



uck-04f03m



uck-04f04m

Krinkles®

Krinkles®



uck-04f00m



uck-04f01m



uck-04f03m

(Krinkles to Krinkles)



uck-04f05m



uck-04f06m



uck-04f07m

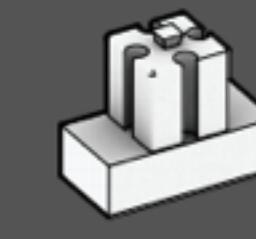


uck-04f08m

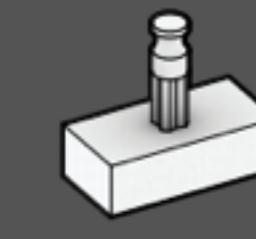


uck-04f09m

(Lego to Duplo
already supported
by manufacturer)



uck-05f01m



uck-05f03m

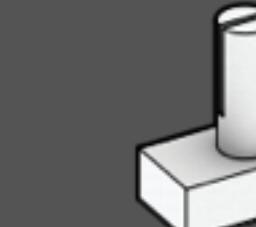


uck-05f04m

(Lego to Lego)



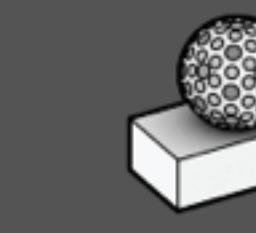
uck-05f06m



uck-05f07m

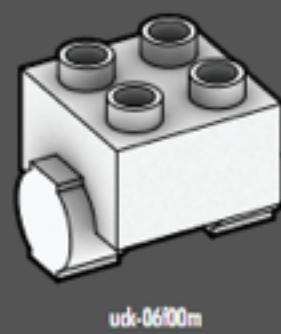


uck-05f08m

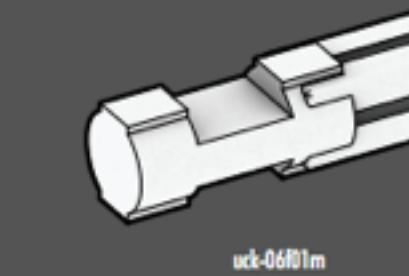


uck-05f09m

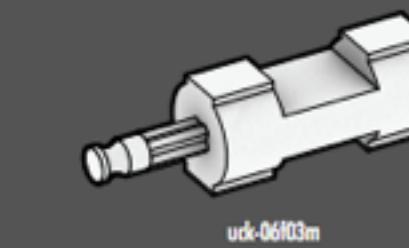
Lego®



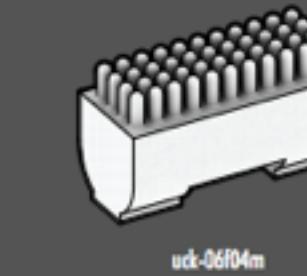
uck-06f00m



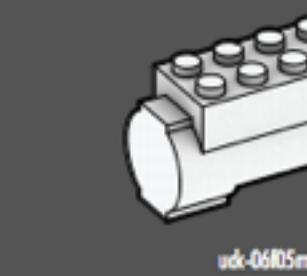
uck-06f01m



uck-06f03m

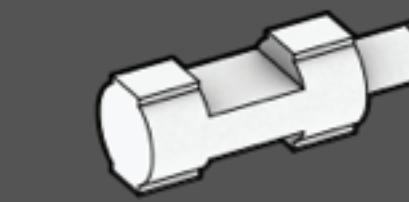


uck-06f04m

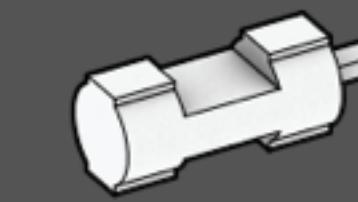


uck-06f05m

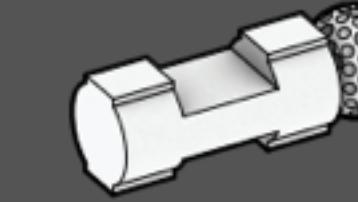
(Lincoln Logs to
Lincoln Logs)



uck-06f07m



uck-06f08m

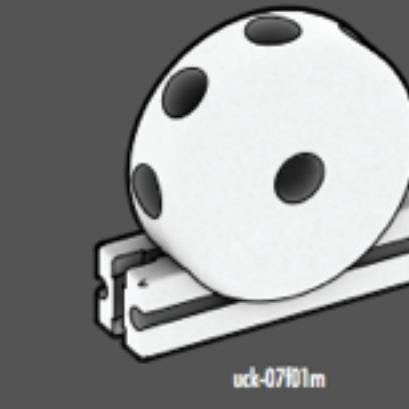


uck-06f09m

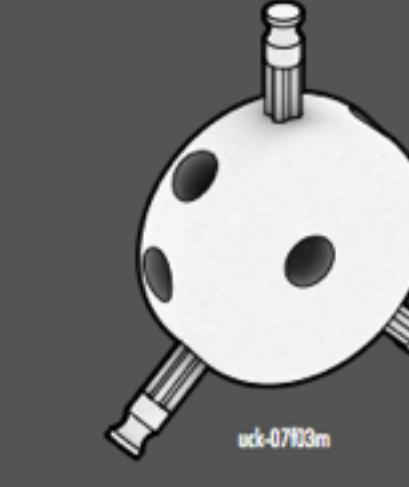
Lincoln Logs®



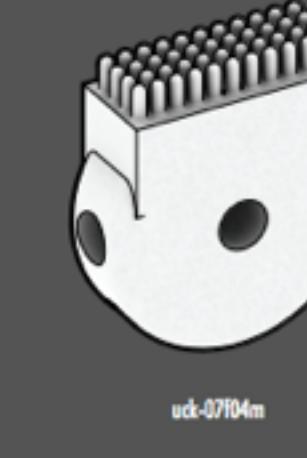
uck-07f00m



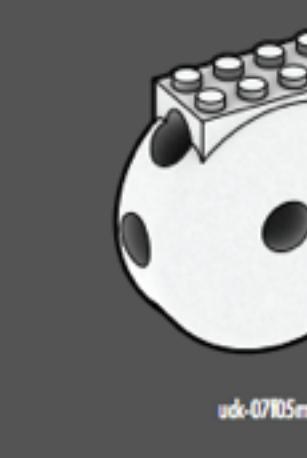
uck-07f01m



uck-07f03m



uck-07f04m



uck-07f05m

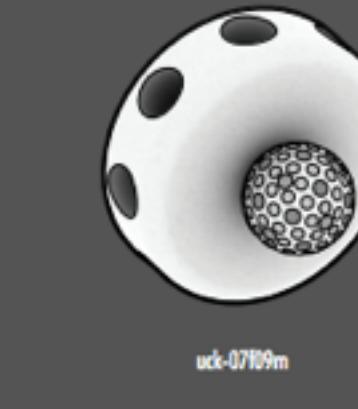


uck-07f06m

(Tinkertoy to
Tinkertoy)



uck-07f08m

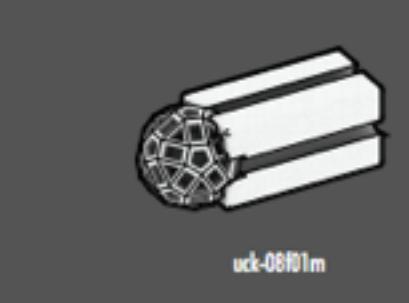


uck-07f09m

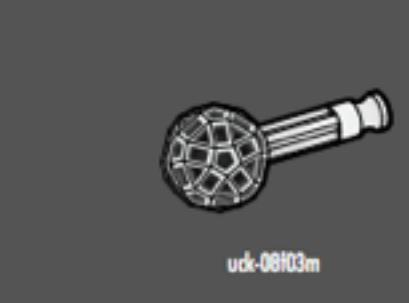
Tinkertoy®



uck-08f00m



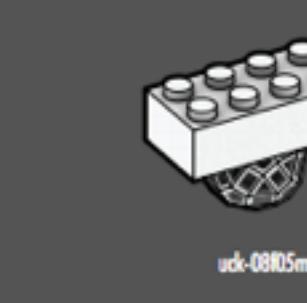
uck-08f01m



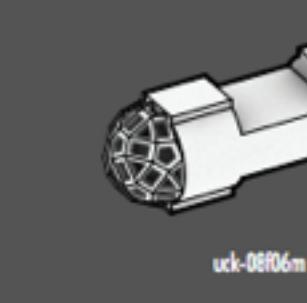
uck-08f03m



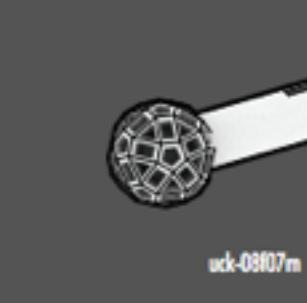
uck-08f04m



uck-08f05m

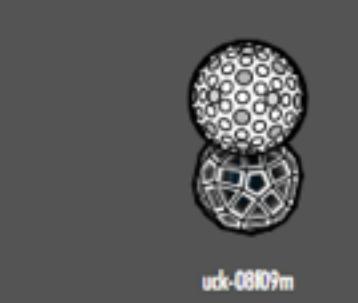


uck-08f06m



uck-08f07m

(ZomeTool to
ZomeTool)

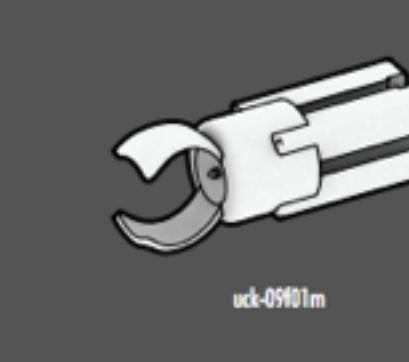


uck-08f09m

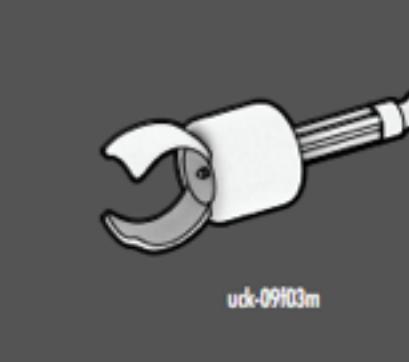
ZomeTool®



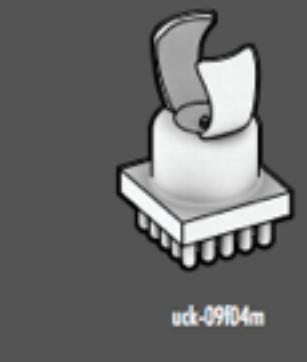
uck-09f00m



uck-09f01m



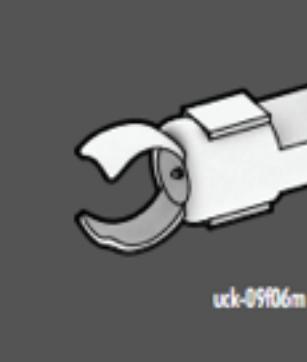
uck-09f03m



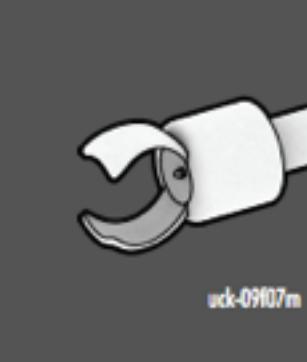
uck-09f04m



uck-09f05m



uck-09f06m



uck-09f07m

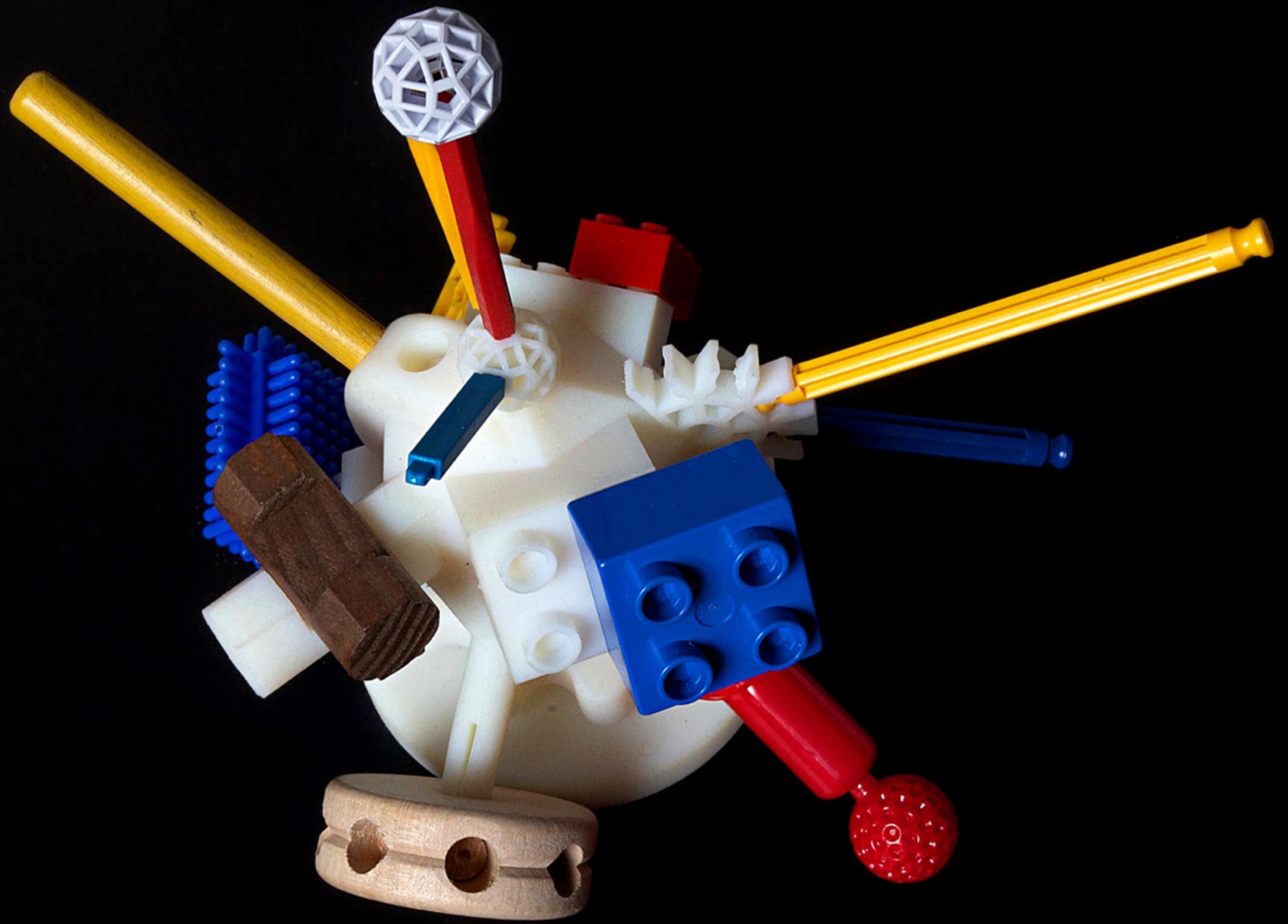


uck-09f08m

(Zoob to Zoob)

Zoob®





<http://fffff.at/free-universal-construction-kit/>

“I thought of objects being like biological cells and/or individual computers on a network, only able to communicate with messages”

—ALAN KAY, writing on OOP

github.com/cognitect/transit-format

- A format & set of elements and extensions for representing typed values.
- Encoded as valid JSON or Msgpack
- Self-describing. Schemas and convention not required.
- Standard libraries for Clojure, JavaScript, Java, Python, and Ruby.

github.com/Prismatic/schema

github.com/intentmedia/schema-transform

- Transform an Avro schema to Prismatic Schema.
- Transform Prismatic to string in Avro format.
- Maintains major primitive types and supports complex types like arrays, maps, and records. Fields can be optional or required.

```
(defschema Record
  {:name      String
   :keywords  [String]
   :updated-at Long})
```

Nulls

- no field at all
- NULL / NIL / NA / NAN
- “”
- “unspecified whitespace”
- “NULL”
- -1 or magic number
- “NOT_FOUND” / “MISSING” / “UNKNOWN”

- Have you weighed performance / convenience? Is it human readable? Does it need to be?
- Are you maintaining type information? Can you maintain data structures?
- Do you have common semantics for boundary cases like NULLs?
- Can you extend and modify your schema safely? Is it versioned?
- Can you extend into new language ecosystems?

- *Easy to use.* Does communicating with other components feel native?
- *Easy to extend.* Can you add new components? New languages? New fields? Without downtime?
- *Easy to generate.* Can we pass around a data blueprint? Automate manual coding?

Fear no data

chetmancini.com

Github/Twitter: @chetmancini

Thanks to these companies:

intentmedia/schema-transform

prismatic/schema

cognitect/transit-format

Free Art and Technology [F.A.T.] Lab and Sy-Lab. “The Free Universal Construction Kit.”

fffff.at/free-universal-construction-kit