The Fun of Connection

Abstraction for Data Systems

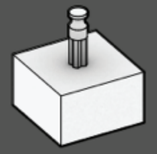
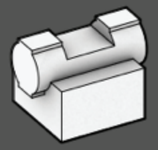
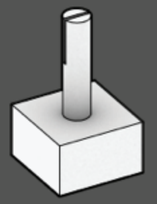
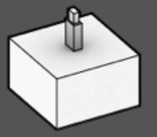
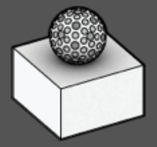
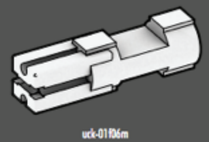
Chet Mancini

Github & Twitter: @chetmancini


LEGO
BASIC
545
120072


6833
LEGOLAND
LEGO
120091


LEGO SYSTEM
6090

Duplo®	Fischertechnik® K'Nex®	Krinkles®	Lego®	Lincoln Logs® Tinkertoy®	ZomeTool®	Zoob®	
Godfred K. Christiansen US Pat. 3597875 / Aug. 10, 1971	Artur Fischer US Pat. 3464147 / Sep. 19, 1966	Joel Glickman & M. Doepner US Pat. 5061219 / Dec. 11, 1990	Walter Haubl US Pat. 3603025 / Sep. 30, 1968	Godtfred K. Christiansen US Pat. 3005282 / Jul. 28, 1958	John Lloyd Wright US Pat. 1351086 / Jan.8, 1920	Charles H. Pajeau US Pat. 1113371 / Jul. 8, 1914	Steven Rogers & P. Hildebrandt US Pat. 6840699 / Nov. 1, 2002*	Michael J. Grey US Pat. 5897417 / Dec. 11, 1996*	
(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®
 uck-04f00m	 uck-04f01m	 uck-04f03m	(Krinkles to Krinkles)	 uck-04f05m	 uck-04f06m	 uck-04f07m	 uck-04f08m	 uck-04f09m	Krinkles®


uck-04100m


uck-04101m


uck-04103m

(Krinkles to Krinkles)


uck-04105m


uck-04106m


uck-04107m

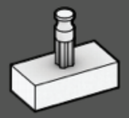

uck-04108m


uck-04109m

Krinkles®

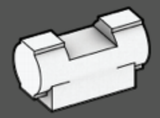
(Lego to Duplo already supported by manufacturer)


uck-05101m

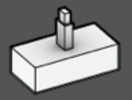

uck-05103m

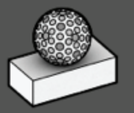

uck-05104m

(Lego to Lego)


uck-05106m

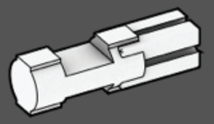

uck-05107m

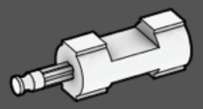

uck-05108m


uck-05109m

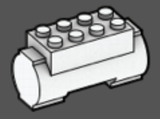
Lego®


uck-06100m

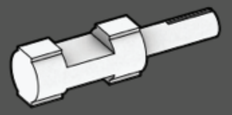

uck-06101m

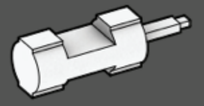

uck-06103m

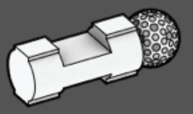

uck-06104m


uck-06105m

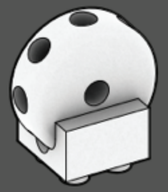
(Lincoln Logs to Lincoln Logs)

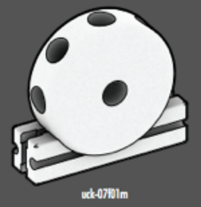

uck-06107m

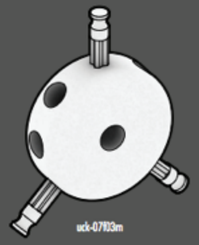

uck-06108m


uck-06109m

Lincoln Logs®


uck-07100m


uck-07101m


uck-07103m

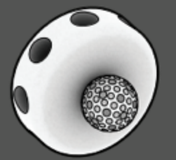

uck-07104m


uck-07105m


uck-07106m

(Tinkertoy to Tinkertoy)


uck-07108m


uck-07109m

Tinkertoy®

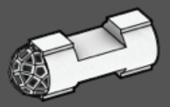

uck-08100m


uck-08101m


uck-08103m

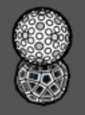

uck-08104m


uck-08105m


uck-08106m


uck-08107m

(ZomeTool to ZomeTool)


uck-08109m

ZomeTool®


uck-09100m

uck-09101m

uck-09103m

uck-09104m

uck-09105m

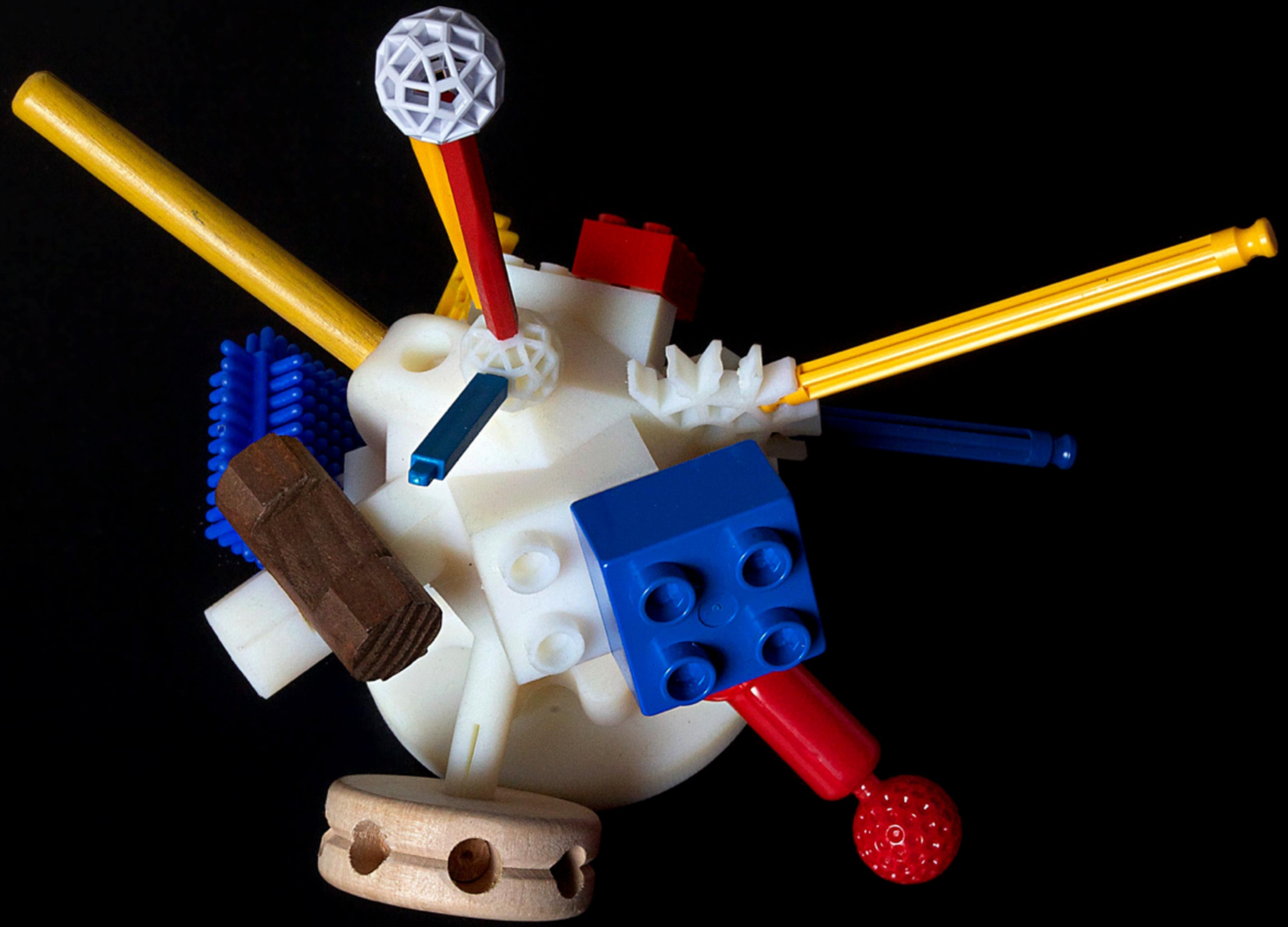
uck-09106m

uck-09107m

uck-09108m

(Zoob to Zoob)

Zoob®



"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