Updates from Austin

2009-08-13

Bryan Bishop

http://heybryan.org/

Ben Lipkowitz

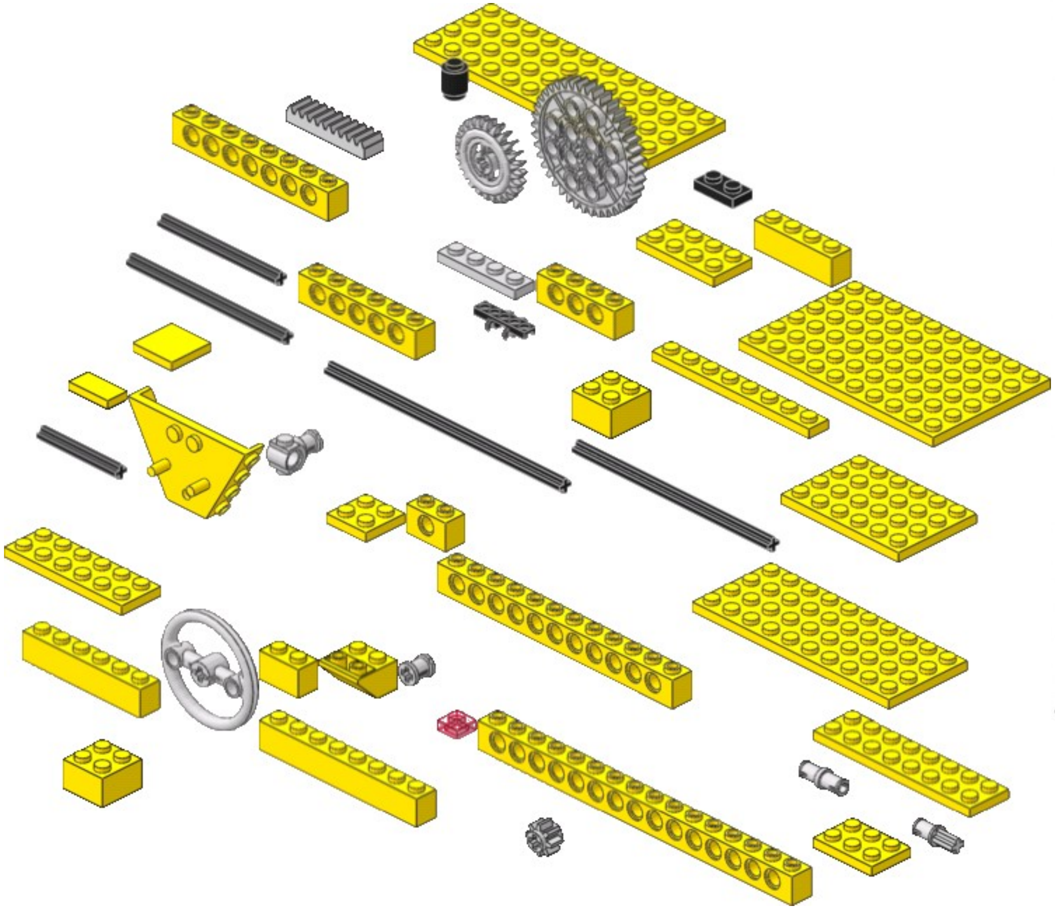
http://fennetic.net/

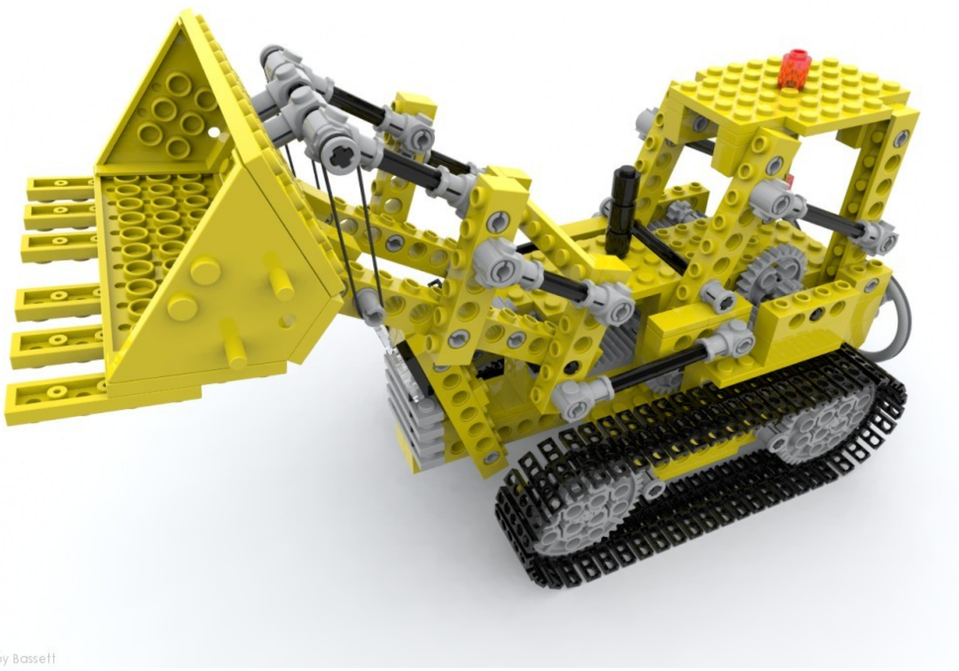
Automated Design Lab at the University of Texas at Austin

Lab wiki:
http://adl.serveftp.org/dokuwiki/

Lab fileserver:
http://adl.serveftp.org/

Lab site:
http://www.me.utexas.edu/~adl/





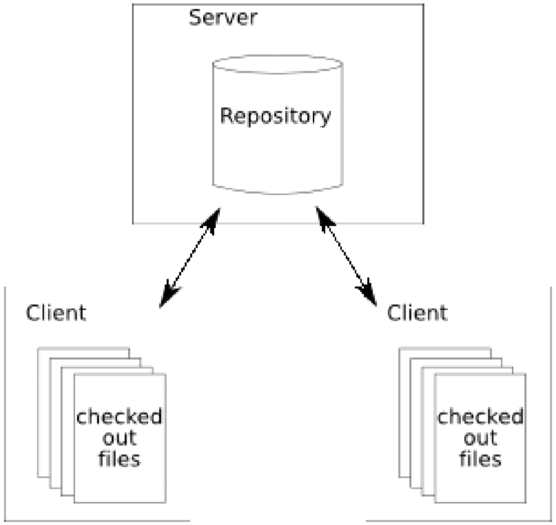
Coby Bassett

National Design Repository

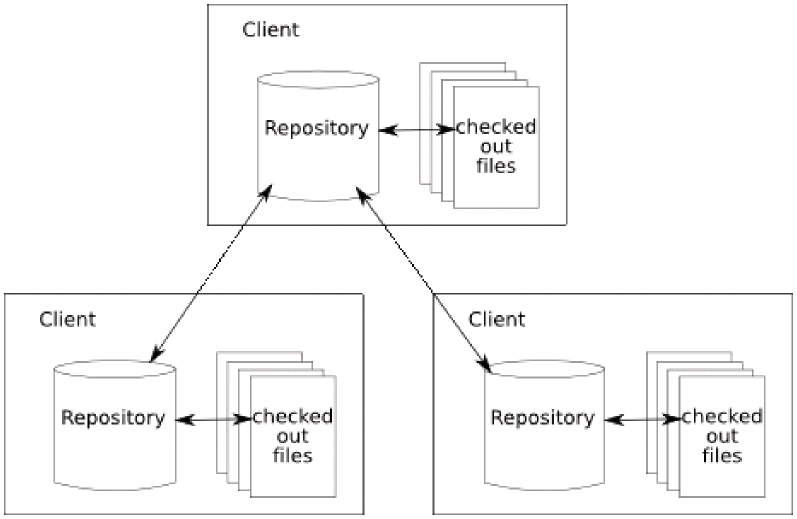
- Over 40,000 CAD files collected in ~2 years
  - Many file formats: IGES, STEP, DXF, SLDPRT, VRML, XML
  - Had a part searching algorithm
- No longer on the internet – blown off the face of the earth
- Long-term viability is important
  - Accessibility (can someone else find it easily?)
  - Share-ability (can a user share/copy information?)
  - Workability (does it work with common tools?)
  - Reliability (is the hardware representation good?)
  - Constructibility (can John Doe make what he finds in the repo?)
- What happens if VOICED and our engineering vanishes too?
- … and what can we do to prevent this?

10 second git intro

Centralized



Distributed



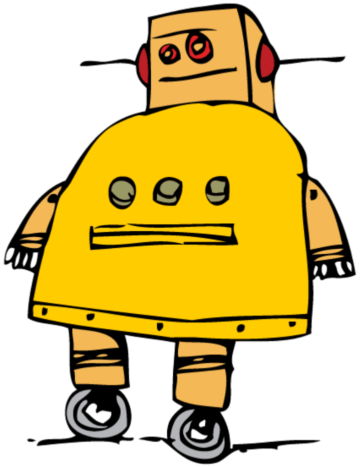
Others popping up on the web

- thingiverse
- instructables
- odesign
- liquidware
- unptnt
- octopart
- ponoko
- shapeways
- OSHbank
- opencores
- openmanufacturing
- diybio
- Pink Army
- skdb (that's us)

Ponoko

Octopart

full fablab inventory



instructables.com

THE WORLD'S BIGGEST SHOW & TELL

Common themes

- Packages
  - Standardized and defined unit of hardware with metadata for distribution
  - “Will hardware ever roam the web like mp3s?” - Dave ten Have, CEO of Ponoko
- DIY (do-it-yourself) and fablabs
- Principle: Always allow the user full control of what is on his machines.
- Overall poor health:
  - instructables.com promotes sending engineering information as photographs (not CAD)
  - Not building off of community progress
  - Isn't a repository supposed to fix this?
    - Software world already went through this (we'll talk about this later)

Proposed User Roles

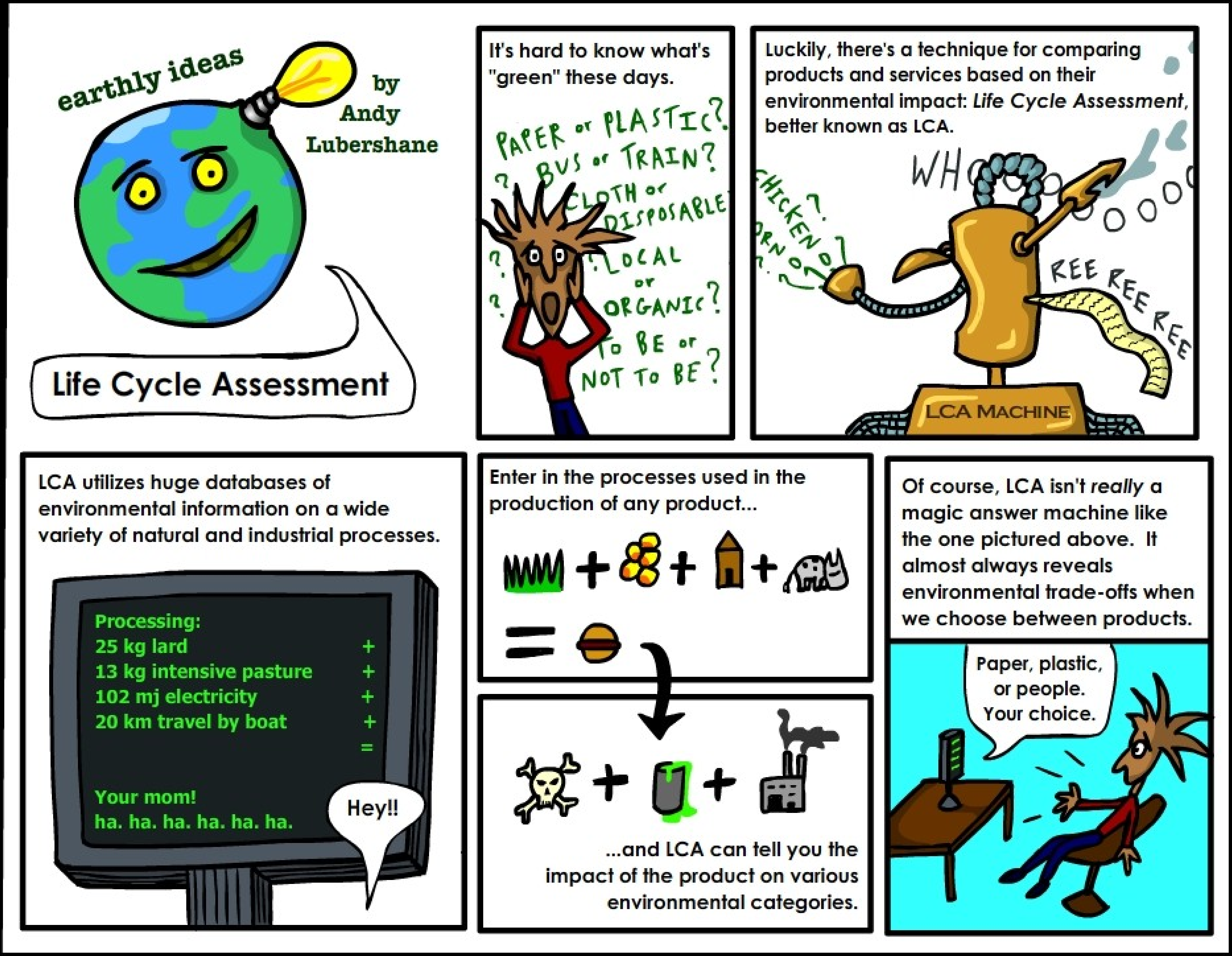
- Trends from the scene might inform academic direction?
- Mutually benefitial relationship between community and engineering academia
- **Makers (users)**
  - Consume content.
  - Build hardware they find interesting or useful.
  - Little or no barrier to entry
  - Ex: anyone
- **Designers/developers**
  - Solve particular problems via design.
  - Don't want to reinvent the wheel.
  - Need to confirm their designs (testing).
  - Evaluate and employ concepts.
  - Ex: programmers, engineers, professors, health care providers, etc. etc.
- **Package maintainers**
  - Knowers of the gnarly details of the system
  - Help users and designers by reviewing designs and making sure nothing breaks the guts of the system.
  - Ex: active debian community
    - “Debian rides the spaceshuttle!” (1997)

User Scenarios

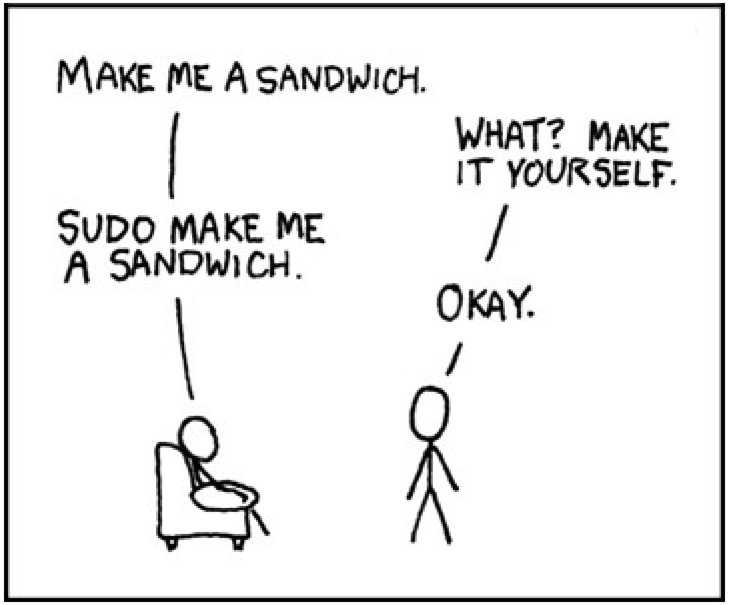
- Mechanical engineering students design a new umbrella and want to offer it as a standard
- Setting up a biolab: what do you need in terms of chemicals and equipment?
- Technician needs a replacement part
- Civil engineer wants to plan city infrastructure
- Someone needs instructions for assembly of a project, or how to carry out a certain procedure.

User Scenarios

- Austin Robot Group members want to submit and package their designs for reuse.
- Dorkbot-Austin builds some PCBs, and collaborate over the internet
- Dr. Freitas wants to build his self replicating lunar factory, but doesn't know where to begin: what does he need to build first?
- Building machine tools from scratch: what machine tool do you start with?


earthly ideas
by Andy Lubershane
Life Cycle Assessment
It's hard to know what's "green" these days.
PAPER or PLASTIC?
BUS or TRAIN?
CLOTH or DISPOSABLE
LOCAL or ORGANIC?
TO BE or NOT TO BE?
Luckily, there's a technique for comparing products and services based on their environmental impact: *Life Cycle Assessment*, better known as LCA.
CHICKEN o?
WHOOOOOOOO
REE REE REE
LCA MACHINE
LCA utilizes huge databases of environmental information on a wide variety of natural and industrial processes.
Processing:
25 kg lard +
13 kg intensive pasture +
102 mj electricity +
20 km travel by boat +
=
Your mom!
ha. ha. ha. ha. ha. ha.
Hey!!
Enter in the processes used in the production of any product...
+ + + =
+ +
...and LCA can tell you the impact of the product on various environmental categories.
Of course, LCA isn't *really* a magic answer machine like the one pictured above. It almost always reveals environmental trade-offs when we choose between products.
Paper, plastic, or people. Your choice.

Published by worldchanging.com



http://boingboing.net/2009/02/27/sudo-make-me-a-sandw.html

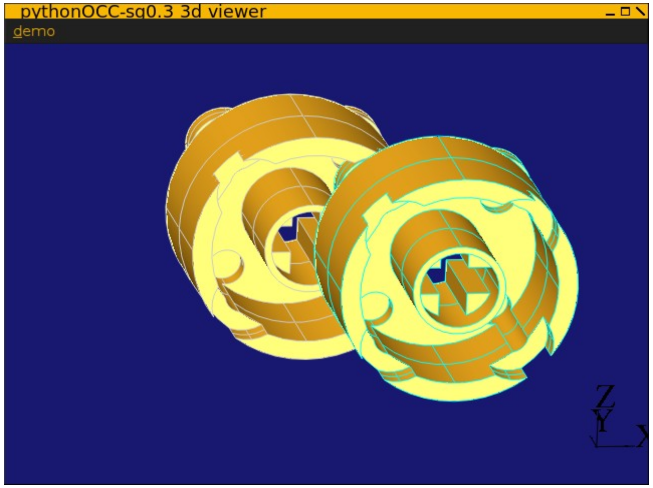
Automated Design Lab Infrastructure and Toolchains

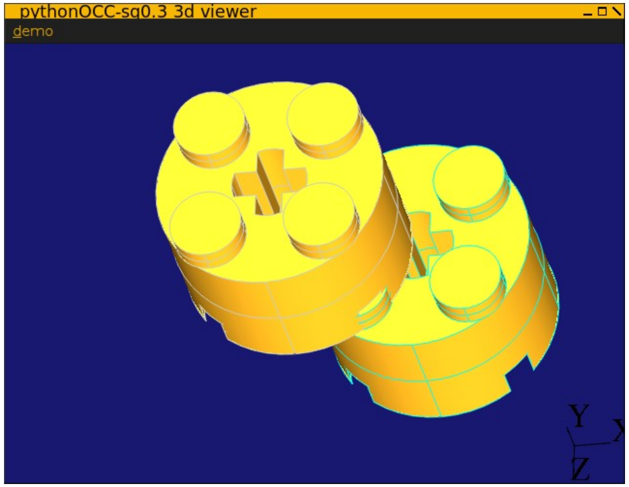
How did the programmers solve their growing pains with the internet?

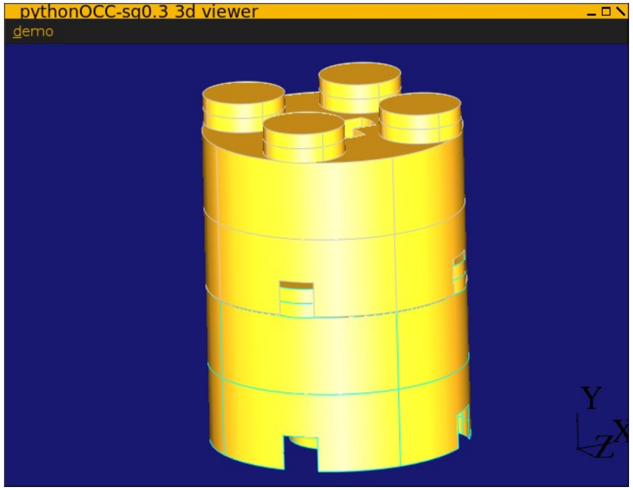
- Version control systems for software
  - Why not hardware too?
- Package management systems for software
  - Why not hardware too?
  - apt-get & dpkg (among others, i.e. portage)
    - How it works
  -
- Autoproject tools (“make”)
- Use highly-available commodity tools in toolchain, but don't restrict options
  - Bryan happens to like: vim, git, diff, uzbl, wget, latex, gnuplot, python, totem
  - But Ben likes nano instead of vim, and mplayer instead of totem
-

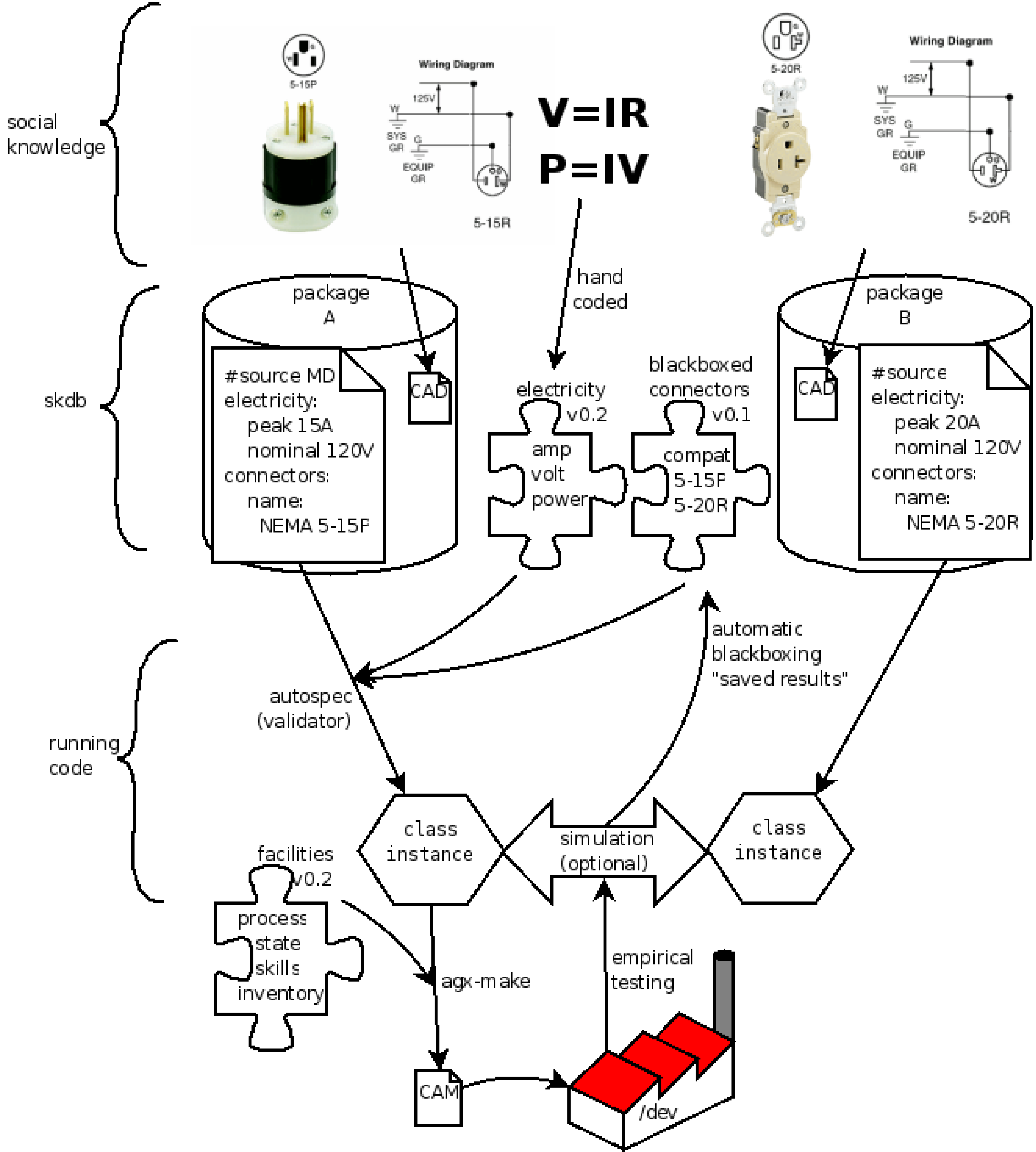
What we've been up to (skdb)

- Working code:
  - Hardware packaging format
  - Part interoperability, compatibility, mating
  - Packages: lego, screw, thread, bearing
- CAD kernel (OpenCASCADE) integration









social knowledge

skdb

running code

V=IR
P=IV

hand coded

package A

#source MD
electricity:
peak 15A
nominal 120V
connectors:
name:
NEMA 5-15P

CAD

electricity v0.2
amp
volt
power

blackboxed connectors v0.1
compat
5-15P
5-20R

package B

CAD

#source
electricity:
peak 20A
nominal 120V
connectors:
name:
NEMA 5-20R

autospec (validator)

automatic blackboxing "saved results"

class instance

simulation (optional)

class instance

facilities v0.2
process
state
skills
inventory

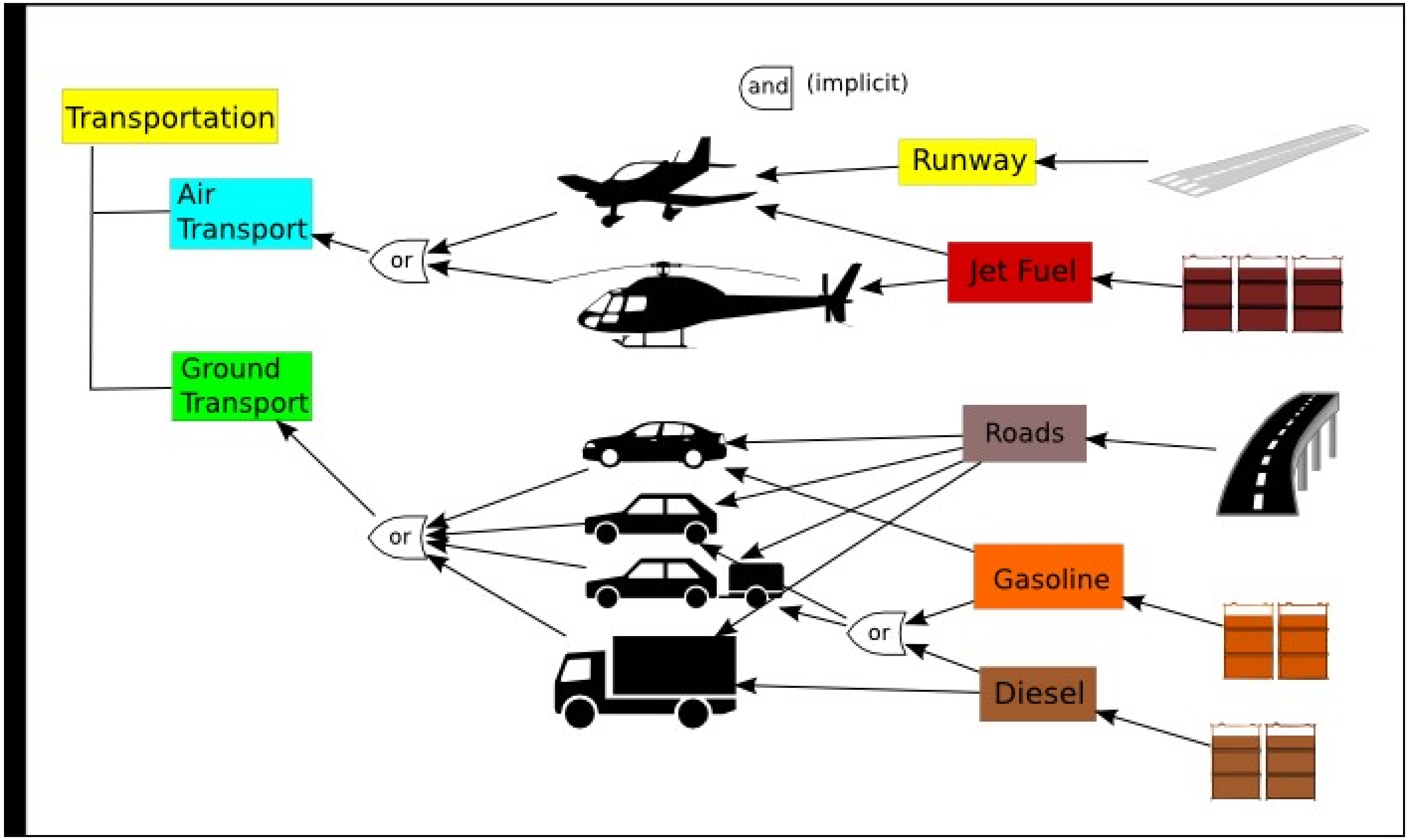
agx-make

empirical testing

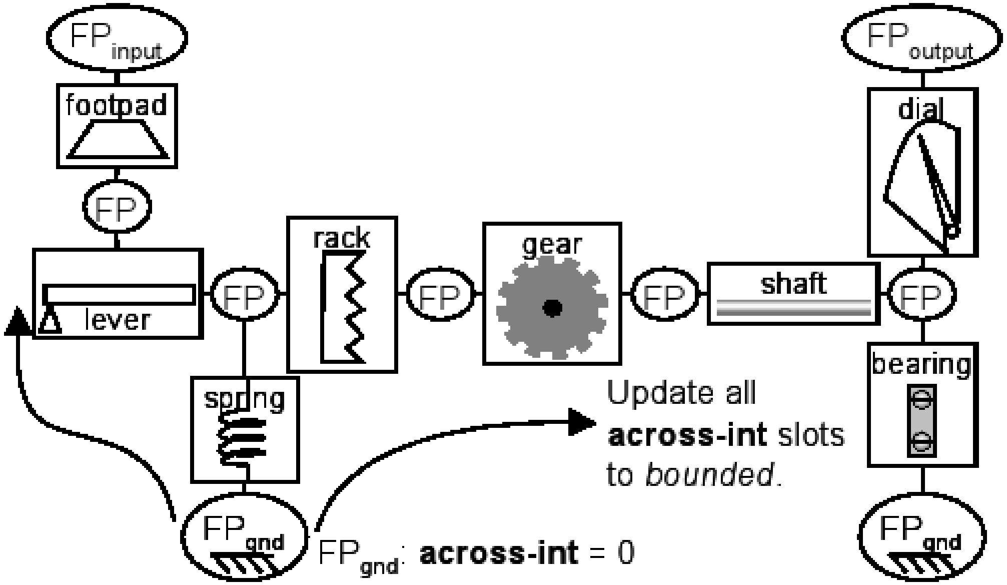
CAM

/dev

Dependency Trees (tech trees)



analysis



Update all **across-int** slots to *bounded*.

FP$_{gnd}$: **across-int** = 0

FP$_{input}$:

Through	{[0 300]}
Across-int	{0}
Across	*bounded*
Across-diff	nil
Class	power
Domain	trans
Interface	feet
Direction	source

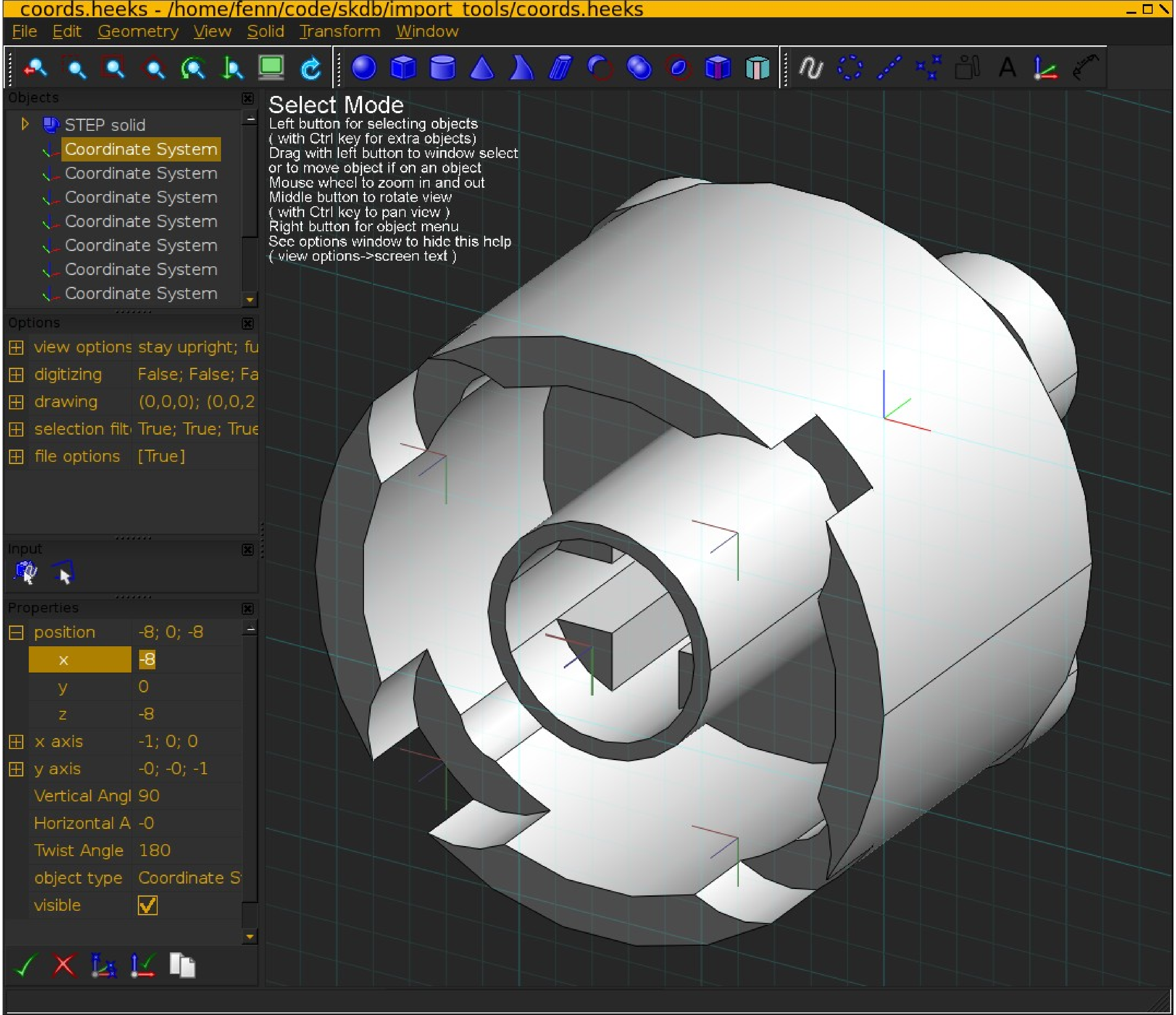
FP$_{output}$:

Through	*bounded*
Across-int	{[0 5]}
Across	*bounded*
Across-diff	nil
Class	power
Domain	rotate
Interface	dial
Direction	sink

$$\theta_{dial}=\frac{1 \qquad d_1}{r_{gear}k_{spring}(d_1+d_2)}F_{weight}$$

$$x_{input}=\frac{1}{k_{spring}}\frac{d_1^2}{(d_1+d_2)^2}F_{weight}$$

HeeksCAD integration



coords.heeks - /home/fenn/code/skdb/import_tools/coords.heeks

- STEP solid
- Coordinate System
- Coordinate System
- Coordinate System
- Coordinate System
- Coordinate System
- Coordinate System
- Coordinate System

Select Mode
Left button for selecting objects
(with Ctrl key for extra objects)
Drag with left button to window select
or to move object if on an object
Mouse wheel to zoom in and out
Middle button to rotate view
(with Ctrl key to pan view)
Right button for object menu
See options window to hide this help
(view options->screen text)

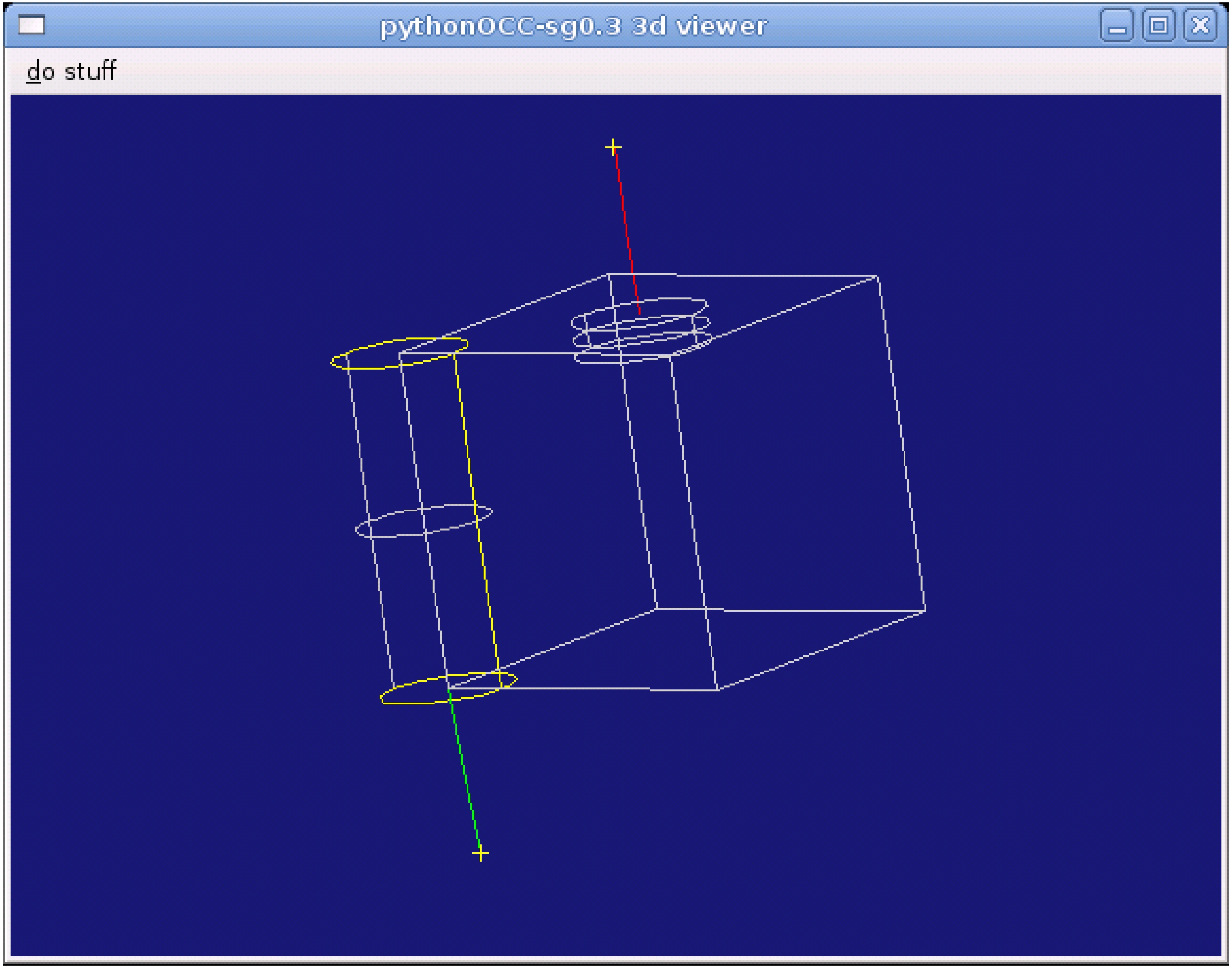
Options

view options	stay upright; fu
digitizing	False; False; Fa
drawing	(0,0,0); (0,0,2
selection filt	True; True; True
file options	[True]

Properties

position	-8; 0; -8
x	-8
y	0
z	-8
x axis	-1; 0; 0
y axis	-0; -0; -1
Vertical Angl	90
Horizontal A	-0
Twist Angle	180
object type	Coordinate S
visible	☑

Part Interfaces



Part Compatibility

- Not quite there yet
- Geometry tags & grammar
- BRep considerations:
    - slop & play
    - volume interference
    - collision detection
- Previous ADL research fitted part compatibility to a probability distribution curve

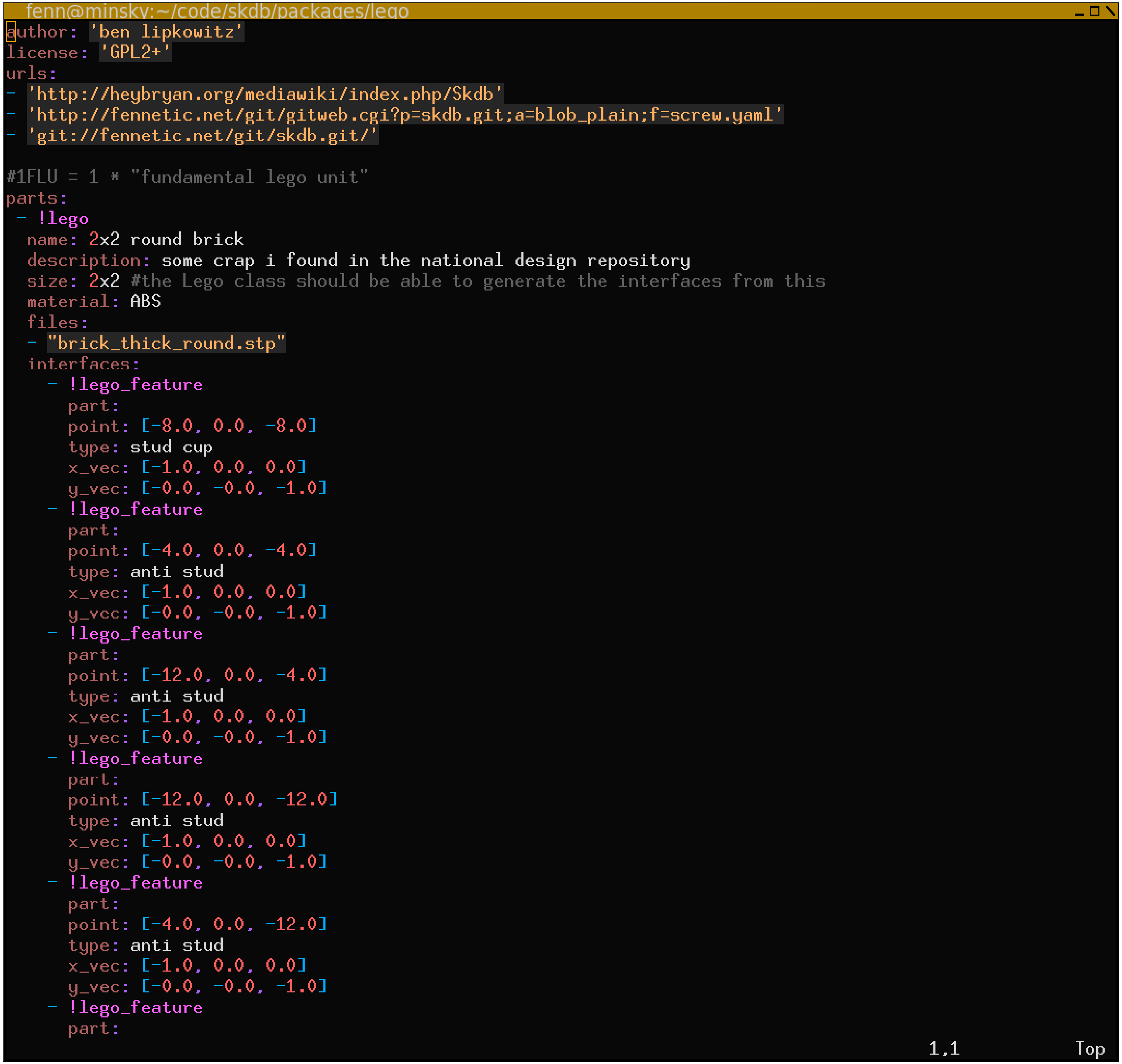
10 second YAML intro

```
foo:
- humpty
- dumpty
- grumpty

bar: 123

myObject: !someclass
  attribute1:
    nested data: [1, 2, 3]
  attribute2:
  attribute3:
```

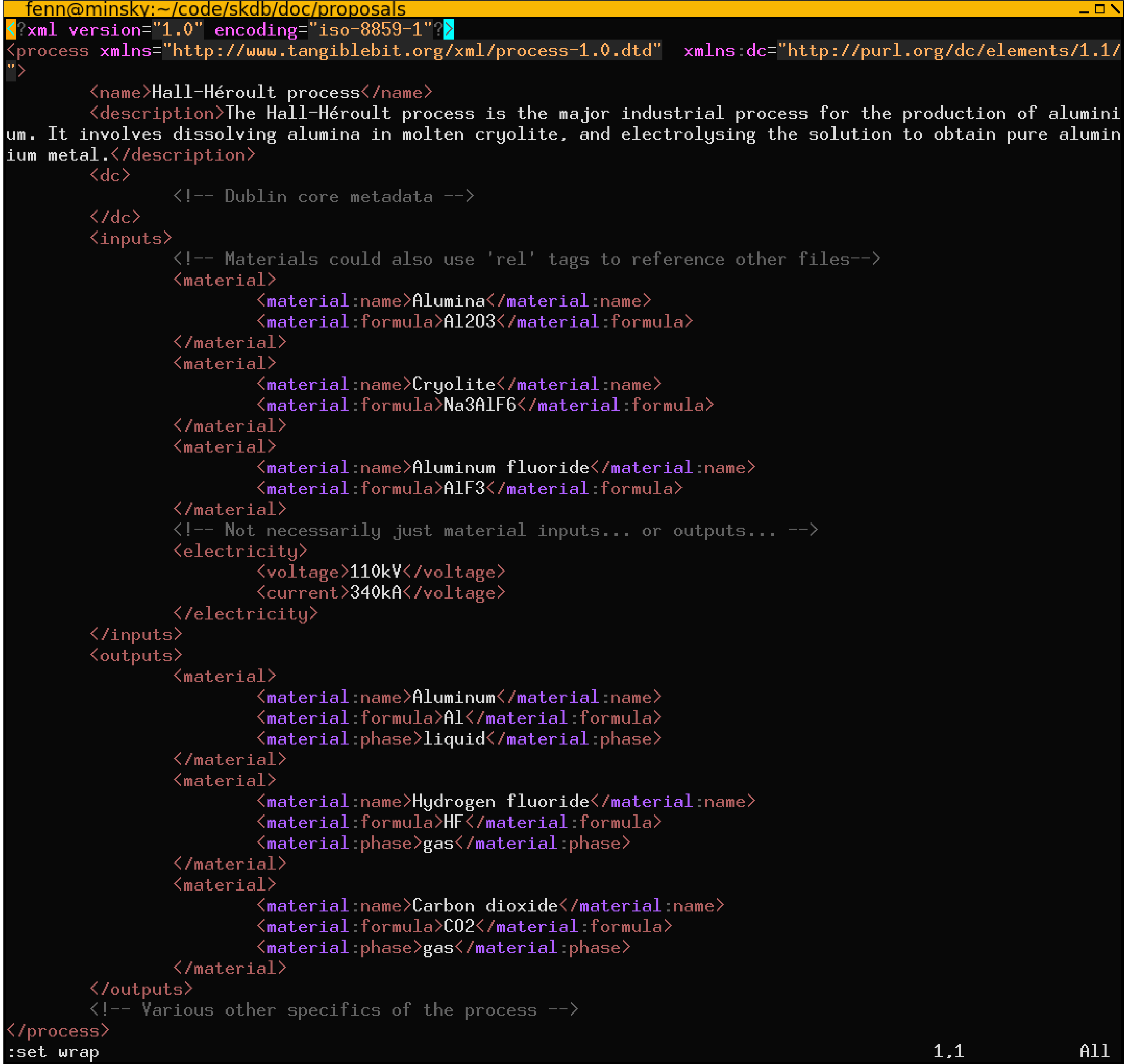
YAML data examples:

- Hardware package metadata (authors, interfaces, etc.)
- Manufacturing process representation
- Catalog data

Actual lego YAML data

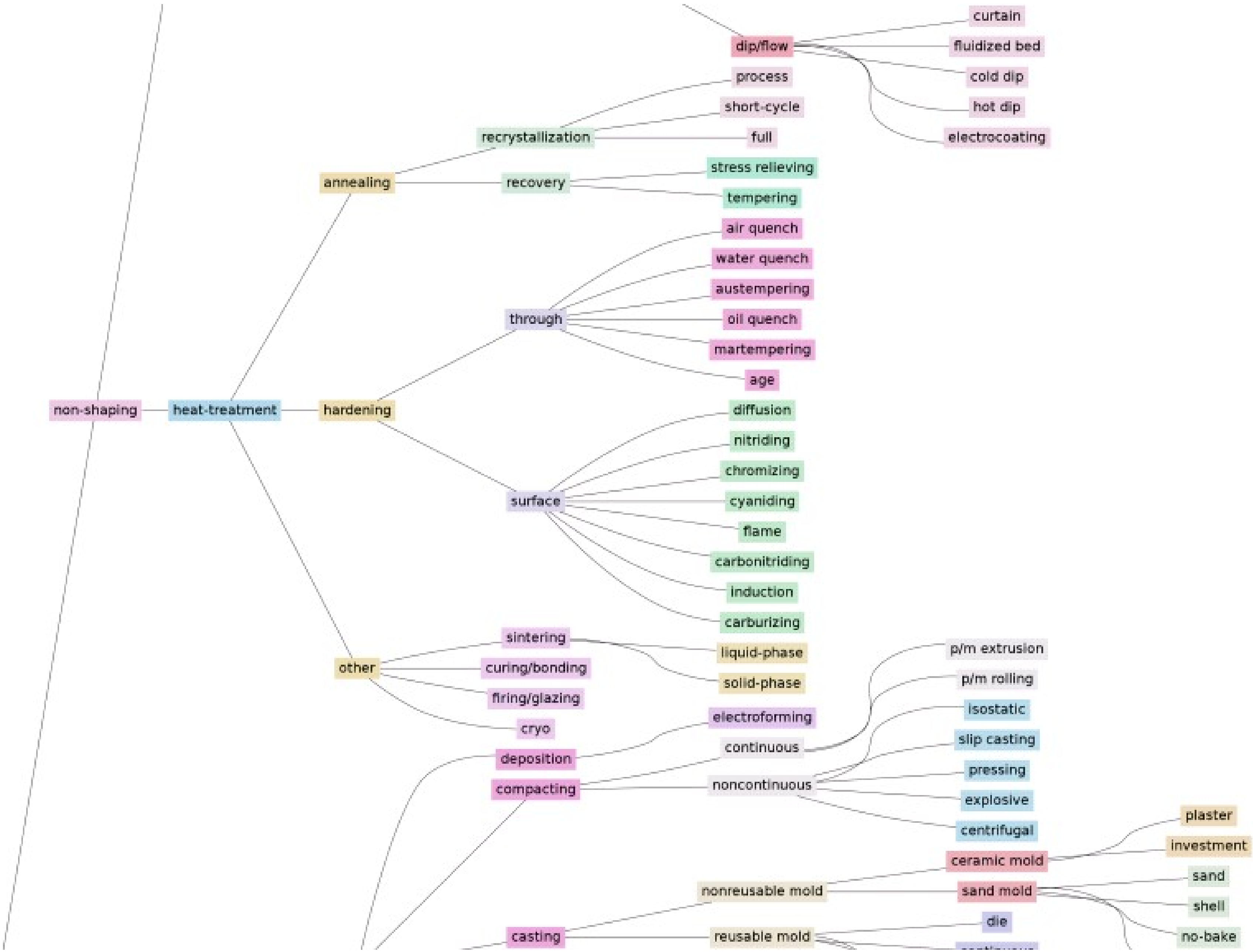


```
author: 'ben lipkowitz'
license: 'GPL2+'
urls:
- 'http://heybryan.org/mediawiki/index.php/Skdb'
- 'http://fennetic.net/git/gitweb.cgi?p=skdb.git;a=blob_plain;f=screw.yaml'
- 'git://fennetic.net/git/skdb.git/'

#1FLU = 1 * "fundamental lego unit"
parts:
 - !lego
  name: 2x2 round brick
  description: some crap i found in the national design repository
  size: 2x2 #the Lego class should be able to generate the interfaces from this
  material: ABS
  files:
  - "brick_thick_round.stp"
  interfaces:
    - !lego_feature
      part:
      point: [-8.0, 0.0, -8.0]
      type: stud cup
      x_vec: [-1.0, 0.0, 0.0]
      y_vec: [-0.0, -0.0, -1.0]
    - !lego_feature
      part:
      point: [-4.0, 0.0, -4.0]
      type: anti stud
      x_vec: [-1.0, 0.0, 0.0]
      y_vec: [-0.0, -0.0, -1.0]
    - !lego_feature
      part:
      point: [-12.0, 0.0, -4.0]
      type: anti stud
      x_vec: [-1.0, 0.0, 0.0]
      y_vec: [-0.0, -0.0, -1.0]
    - !lego_feature
      part:
      point: [-12.0, 0.0, -12.0]
      type: anti stud
      x_vec: [-1.0, 0.0, 0.0]
      y_vec: [-0.0, -0.0, -1.0]
    - !lego_feature
      part:
      point: [-4.0, 0.0, -12.0]
      type: anti stud
      x_vec: [-1.0, 0.0, 0.0]
      y_vec: [-0.0, -0.0, -1.0]
    - !lego_feature
      part:
                                                                                                          1,1           Top
```



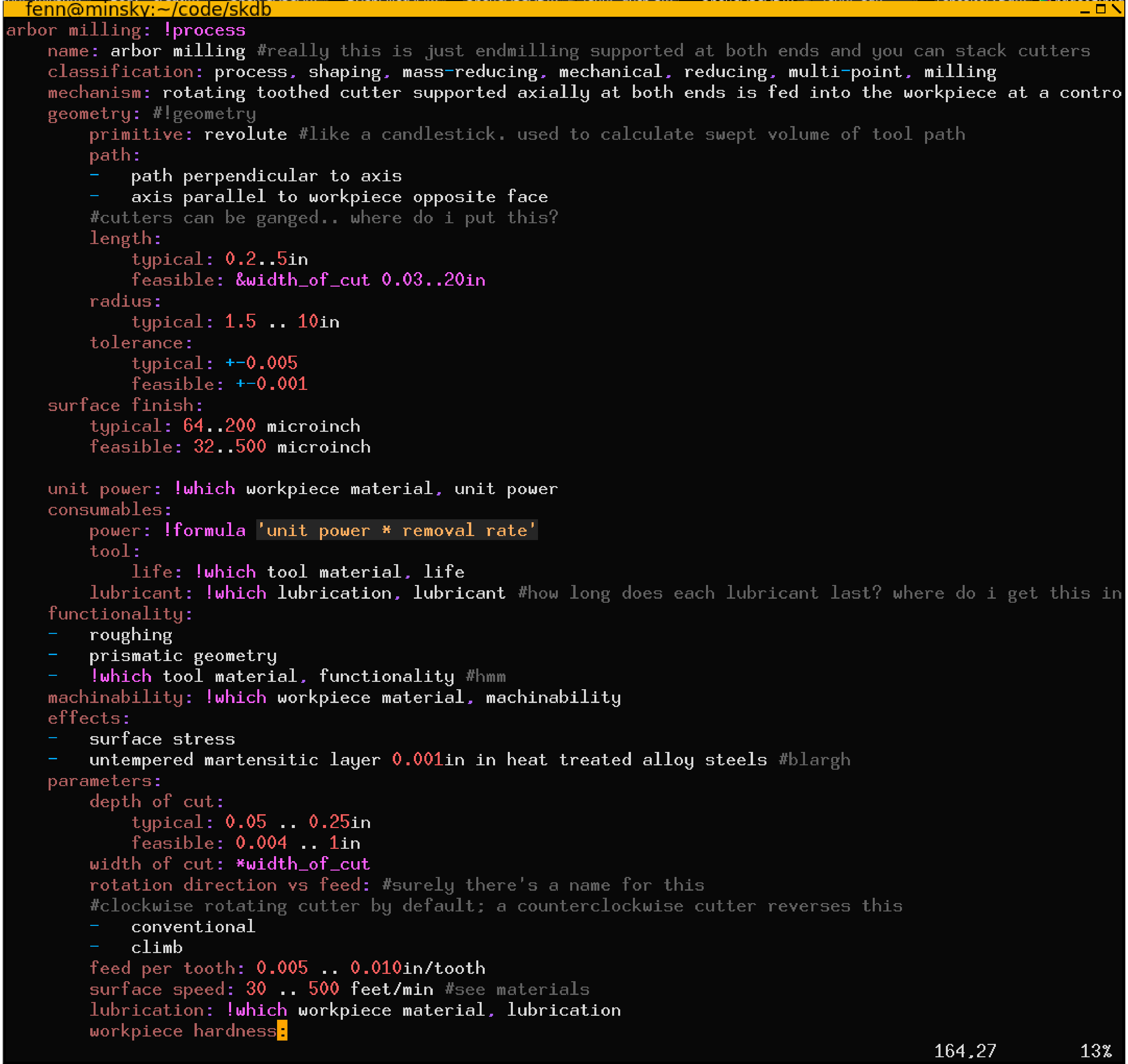
```
<?xml version="1.0" encoding="iso-8859-1"?>
<process xmlns="http://www.tangiblebit.org/xml/process-1.0.dtd"  xmlns:dc="http://purl.org/dc/elements/1.1/">
        <name>Hall-Héroult process</name>
        <description>The Hall-Héroult process is the major industrial process for the production of aluminium. It involves dissolving alumina in molten cryolite, and electrolysing the solution to obtain pure aluminium metal.</description>
        <dc>
                <!-- Dublin core metadata -->
        </dc>
        <inputs>
                <!-- Materials could also use 'rel' tags to reference other files-->
                <material>
                        <material:name>Alumina</material:name>
                        <material:formula>Al2O3</material:formula>
                </material>
                <material>
                        <material:name>Cryolite</material:name>
                        <material:formula>Na3AlF6</material:formula>
                </material>
                <material>
                        <material:name>Aluminum fluoride</material:name>
                        <material:formula>AlF3</material:formula>
                </material>
                <!-- Not necessarily just material inputs... or outputs... -->
                <electricity>
                        <voltage>110kV</voltage>
                        <current>340kA</voltage>
                </electricity>
        </inputs>
        <outputs>
                <material>
                        <material:name>Aluminum</material:name>
                        <material:formula>Al</material:formula>
                        <material:phase>liquid</material:phase>
                </material>
                <material>
                        <material:name>Hydrogen fluoride</material:name>
                        <material:formula>HF</material:formula>
                        <material:phase>gas</material:phase>
                </material>
                <material>
                        <material:name>Carbon dioxide</material:name>
                        <material:formula>CO2</material:formula>
                        <material:phase>gas</material:phase>
                </material>
        </outputs>
        <!-- Various other specifics of the process -->
</process>
:set wrap                                                              1,1           All
```

Manufacturing Process Taxonomy

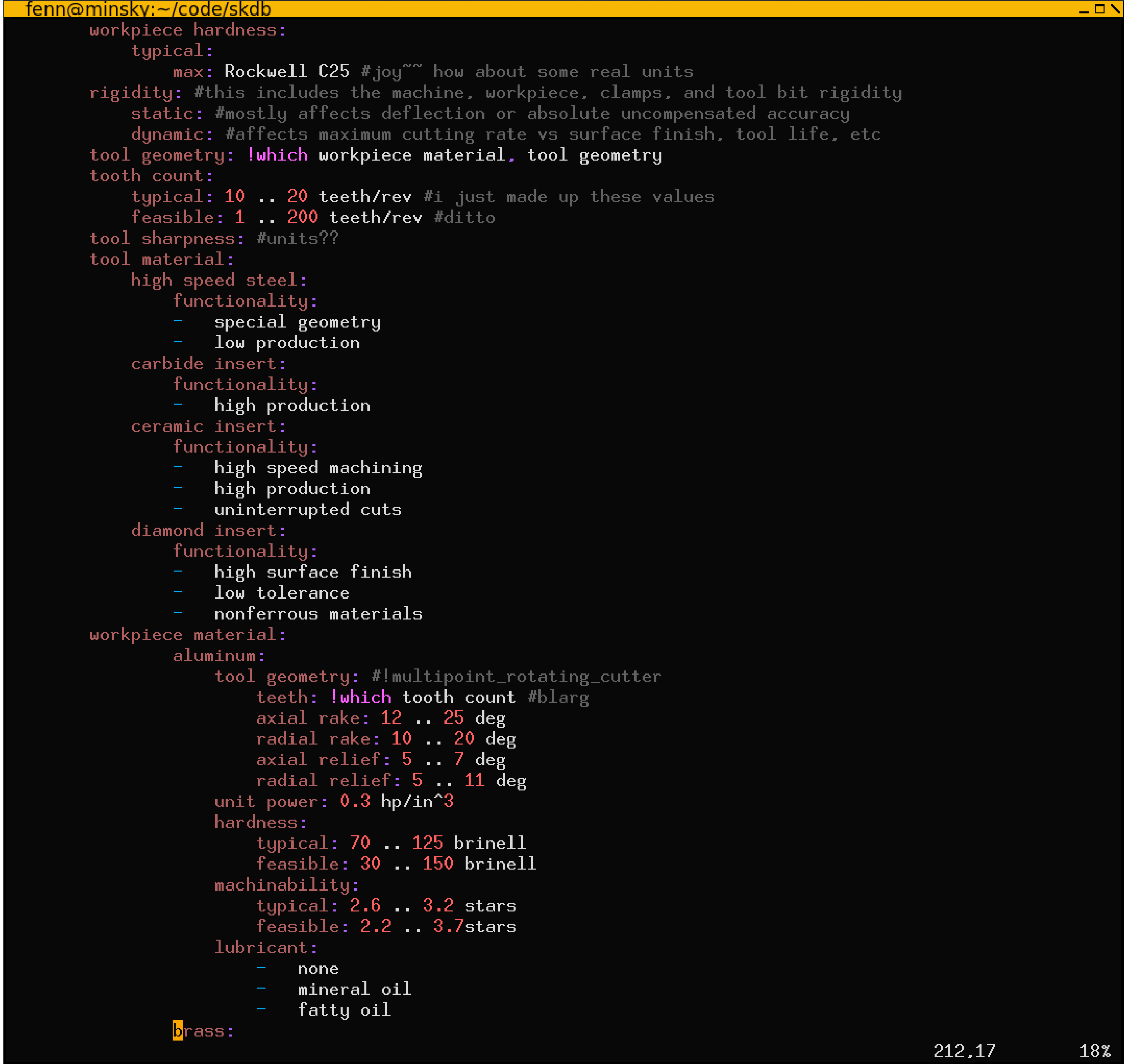


Manufacturing Process Representation

- YAML example on next slide
- Process is what happens to matter, energy and information
- “A process can be carried out by hand or by machine.”
- Wanted: general geometry constraint language. Does it exist?

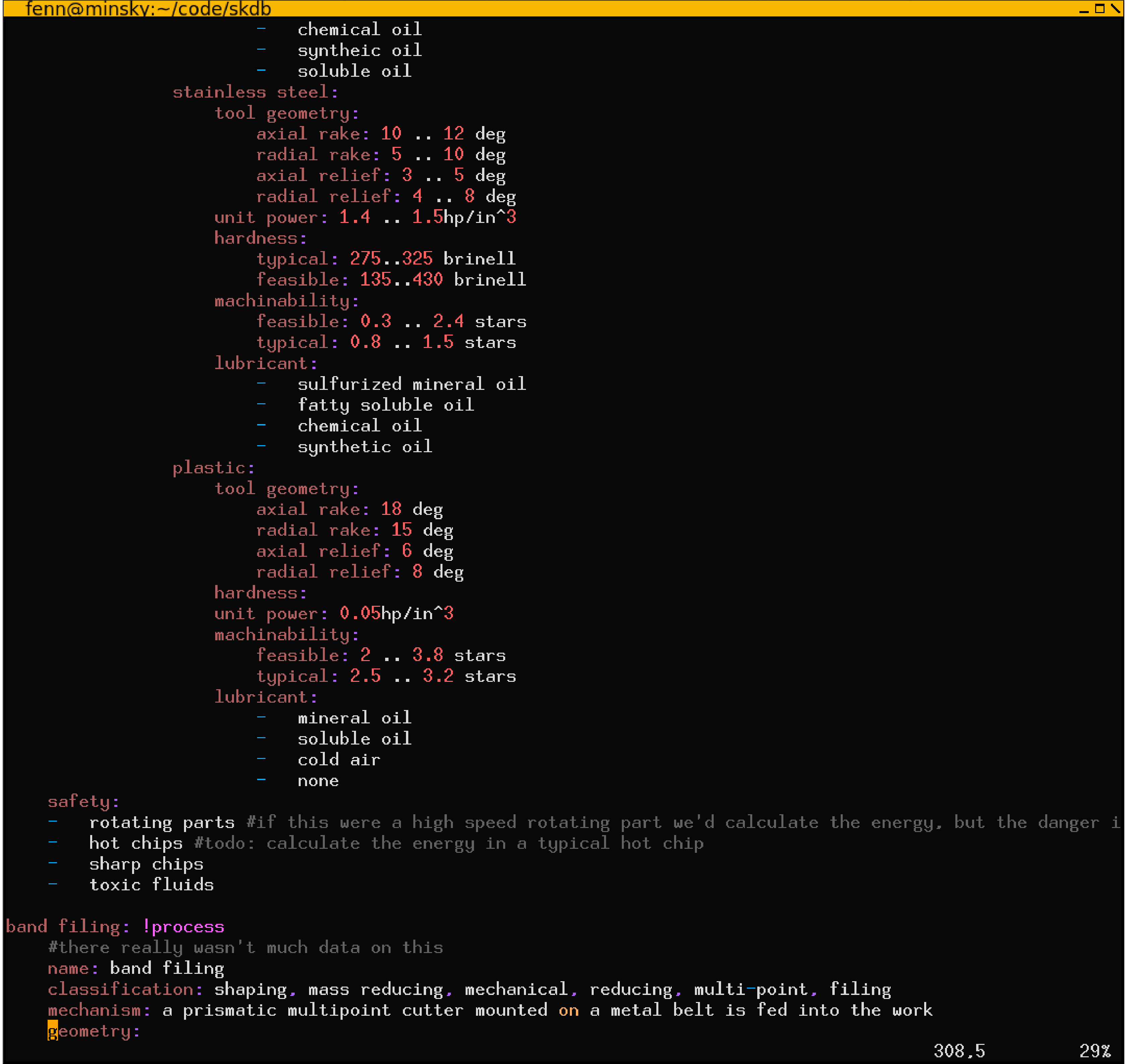


```
arbor milling: !process
    name: arbor milling #really this is just endmilling supported at both ends and you can stack cutters
    classification: process, shaping, mass-reducing, mechanical, reducing, multi-point, milling
    mechanism: rotating toothed cutter supported axially at both ends is fed into the workpiece at a contro
    geometry: #!geometry
        primitive: revolute #like a candlestick. used to calculate swept volume of tool path
        path:
        -   path perpendicular to axis
        -   axis parallel to workpiece opposite face
        #cutters can be ganged.. where do i put this?
        length:
            typical: 0.2..5in
            feasible: &width_of_cut 0.03..20in
        radius:
            typical: 1.5 .. 10in
        tolerance:
            typical: +-0.005
            feasible: +-0.001
    surface finish:
        typical: 64..200 microinch
        feasible: 32..500 microinch

    unit power: !which workpiece material, unit power
    consumables:
        power: !formula 'unit power * removal rate'
        tool:
            life: !which tool material, life
        lubricant: !which lubrication, lubricant #how long does each lubricant last? where do i get this in
    functionality:
    -   roughing
    -   prismatic geometry
    -   !which tool material, functionality #hmm
    machinability: !which workpiece material, machinability
    effects:
    -   surface stress
    -   untempered martensitic layer 0.001in in heat treated alloy steels #blargh
    parameters:
        depth of cut:
            typical: 0.05 .. 0.25in
            feasible: 0.004 .. 1in
        width of cut: *width_of_cut
        rotation direction vs feed: #surely there's a name for this
        #clockwise rotating cutter by default; a counterclockwise cutter reverses this
        -   conventional
        -   climb
        feed per tooth: 0.005 .. 0.010in/tooth
        surface speed: 30 .. 500 feet/min #see materials
        lubrication: !which workpiece material, lubrication
        workpiece hardness:
                                                                                        164,27        13%
```



fenn@minsky:~/code/skdb

```
        workpiece hardness:
            typical:
                max: Rockwell C25 #joy~~ how about some real units
        rigidity: #this includes the machine, workpiece, clamps, and tool bit rigidity
            static: #mostly affects deflection or absolute uncompensated accuracy
            dynamic: #affects maximum cutting rate vs surface finish, tool life, etc
        tool geometry: !which workpiece material, tool geometry
        tooth count:
            typical: 10 .. 20 teeth/rev #i just made up these values
            feasible: 1 .. 200 teeth/rev #ditto
        tool sharpness: #units??
        tool material:
            high speed steel:
                functionality:
                -   special geometry
                -   low production
            carbide insert:
                functionality:
                -   high production
            ceramic insert:
                functionality:
                -   high speed machining
                -   high production
                -   uninterrupted cuts
            diamond insert:
                functionality:
                -   high surface finish
                -   low tolerance
                -   nonferrous materials
        workpiece material:
                aluminum:
                    tool geometry: #!multipoint_rotating_cutter
                        teeth: !which tooth count #blarg
                        axial rake: 12 .. 25 deg
                        radial rake: 10 .. 20 deg
                        axial relief: 5 .. 7 deg
                        radial relief: 5 .. 11 deg
                    unit power: 0.3 hp/in^3
                    hardness:
                        typical: 70 .. 125 brinell
                        feasible: 30 .. 150 brinell
                    machinability:
                        typical: 2.6 .. 3.2 stars
                        feasible: 2.2 .. 3.7stars
                    lubricant:
                        -   none
                        -   mineral oil
                        -   fatty oil
                brass:
                                                                                            212,17        18%
```



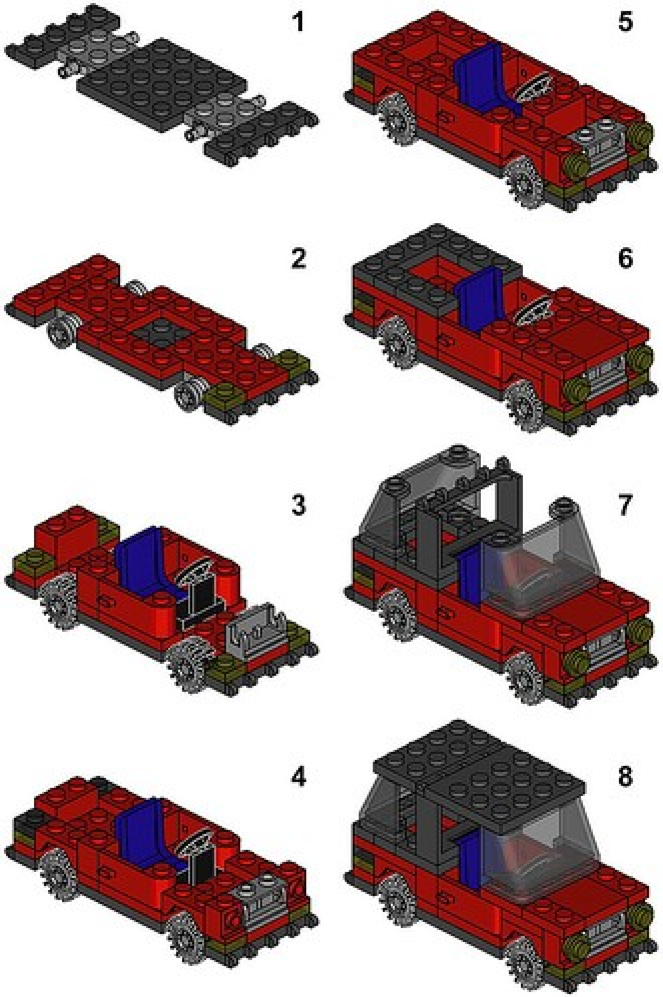
```
                        -   chemical oil
                        -   syntheic oil
                        -   soluble oil
                stainless steel:
                    tool geometry:
                        axial rake: 10 .. 12 deg
                        radial rake: 5 .. 10 deg
                        axial relief: 3 .. 5 deg
                        radial relief: 4 .. 8 deg
                    unit power: 1.4 .. 1.5hp/in^3
                    hardness:
                        typical: 275..325 brinell
                        feasible: 135..430 brinell
                    machinability:
                        feasible: 0.3 .. 2.4 stars
                        typical: 0.8 .. 1.5 stars
                    lubricant:
                        -   sulfurized mineral oil
                        -   fatty soluble oil
                        -   chemical oil
                        -   synthetic oil
                plastic:
                    tool geometry:
                        axial rake: 18 deg
                        radial rake: 15 deg
                        axial relief: 6 deg
                        radial relief: 8 deg
                    hardness:
                    unit power: 0.05hp/in^3
                    machinability:
                        feasible: 2 .. 3.8 stars
                        typical: 2.5 .. 3.2 stars
                    lubricant:
                        -   mineral oil
                        -   soluble oil
                        -   cold air
                        -   none
    safety:
    -   rotating parts #if this were a high speed rotating part we'd calculate the energy, but the danger i
    -   hot chips #todo: calculate the energy in a typical hot chip
    -   sharp chips
    -   toxic fluids

band filing: !process
    #there really wasn't much data on this
    name: band filing
    classification: shaping, mass reducing, mechanical, reducing, multi-point, filing
    mechanism: a prismatic multipoint cutter mounted on a metal belt is fed into the work
    geometry:
                                                                                        308,5         29%
```

Manufacturing Processes

- Different packages implement different techniques
  - Milling technique utilizes machining process
  - Milling machine implements milling technique
  - Different milling machines have slightly different ranges for parameters to the milling technique
  - But in general they all follow the same technique
- Can technique generate my geometry?
  - Volumetric sweeps
- Plan: techniques in skdb should generate both:
  - Human-readable instructions
  - Machine instructions (gcode)

    but never just gcode (why?)

Assemble Designs from Repository

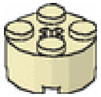


in this case you would use a press fit technique

Web Interface

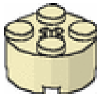
- Allows non-technical users to contribute
- Facilitates browsing and presentation
- Good ideas are out there (next slide)
- Technical details:
  - Anyone can be a developer (without breaking the system)
  - wiki with git-backend
  - django, pylons, pyjamas
    - views: project view, part/CAD summary view
    - YAML easy to edit in browsers
    - Validate user contributions immediately for “common sense”
  - RESTful

3941 Brick 2 x 2 Round



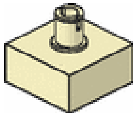
LDraw File: [3941.DAT]
Peeron: Brick 2 x 2 Round
Jessiman: 2 x 2 Round
Vattima: 2 X 2 Round Brick
Patterned Elements: 1 part
See Detail Page

6143 Brick 2 x 2 Round Type 2



LDraw File: [6143.DAT]

4729 Brick 2 x 2 no Studs with Pin

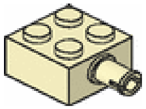


LDraw File: [4729.DAT]
Peeron: Brick 2 x 2 no Studs with Pin
Vattima: 2 X 2 Brick without knobs with male clip joint on top

30165 Brick 2 x 2 with Curved Top and 2 Studs or

LDraw File: [30165.DAT]
Peeron: Brick 2 x 2 with Curved Top and 2 Studs on Top

4730 Brick 2 x 2 with Pin



LDraw File: [4730.DAT]
Peeron: Brick 2 x 2 with Pin
Jessiman: 2 x 2 Brick with Pin

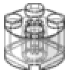
Sets that have 'Brick 2 x 2 Round' (3941) :

Partsref link for 'Brick 2 x 2 Round' [DAT]

See similar elements. Include patterns

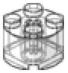
My Parts: ⧗

Black - 505 in 198 sets.
Blue - 151 in 44 sets.
BtGreen - 3 in 2 sets.
ChromeSilver - 16 in 3 sets.
Clear - 18 in 10 sets.
DkRed - 2 in 1 set.
DkStone - 130 in 36 sets.
Green - 5 in 3 sets.
Lime - 24 in 6 sets.
Magenta - 1 in 1 set.
MdStone - 178 in 43 sets.
NavyBlue - 30 in 4 sets.
OldBrown - 152 in 20 sets.
OldGray - 343 in 98 sets.
Orange - 28 in 12 sets.
Red - 221 in 71 sets.
RedBrown - 162 in 26 sets.
SandBlue - 1 in 1 set.
SandGreen - 15 in 3 sets.
Tan - 85 in 19 sets.
TrBlue - 18 in 8 sets.
TrDkOrange - 1 in 1 set.
TrLt Blue - 7 in 2 sets.
TrNeonGreen - 92 in 25 sets.
TrNeonOrange - 59 in 23 sets.
TrRed - 6 in 2 sets.
TrYellow - 86 in 55 sets.
White - 454 in 136 sets.
Yellow - 271 in 82 sets.



Zoom

Additional images:



Piece color in picture may not match colors listed. Printed patterns are correct unless noted.

Black:

- 20 in 6391-1 - Cargo Center (1984)
- 12 in 6990-1 - Monorail Transport System (1987)
- 10 in 4795-1 - Ogel Underwater Base and AT Sub (2002)
- 8 in 9723-1 - Cities and Transportation (2000)
- 8 in 3804-1 - Robotics Invention System 2.0 (2001)
- 8 in 7905-1 - Building Crane (2006)
- 8 in 9794-1 - Team Challenge Set {updated}, with USB cable (2003)
- 8 in 7186-1 - Watto's™ Junkyard (2001)
- 8 in 9747-1 - Robotics Invention System 1.5 (1999)
- 8 in 9719-1 - Robotics Invention System 1.0 (1998)
- 8 in 8160-1 - Cruncher Block and Racer X (2008)
- 7 in 9320-1 - Journey Into Space Set (2003)
- 7 in 8285-2 - Tow Truck (2006)
- 6 in 7180-1 - B-wing™ at Rebel Control Center (2000)
- 6 in 678-1 - Knights' Kingdom Chess Set (2005)
- 6 in 6950-1 - Mobile Rocket Transport (1982)
- 6 in 6497-1 - Twisted Time Train (1997)
- 6 in 6257-1 - Castaway's Raft (1989)
- 6 in 9761-1 - FIRST LEGO League Challenge 2004 - No Limits (2004)
- 6 in 8275-1 - Motorized Bulldozer (2007)
- *178 sets were skipped containing 343 parts.*
  Show all the Black Brick 2 x 2 Round.

Blue:

- 16 in 6930-1 - Space Supply Station (1983)
- 12 in 7727-1 - Electric Freight Train (1983)
- 7 in 4405-1 - Large Creator Bucket (2003)
- 7 in 1782-1 - Discovery Station (1997)
- 6 in 7171-1 - Mos Espa Podrace™ (1999)
- 6 in 483-1 - Alpha-1 Rocket Base (1979)
- 6 in 7778-1 - Midi-scale Millenium Falcon (2009)
- 6 in 7675-1 - AT-TE Walker (2008)

Available from these Associated stores:

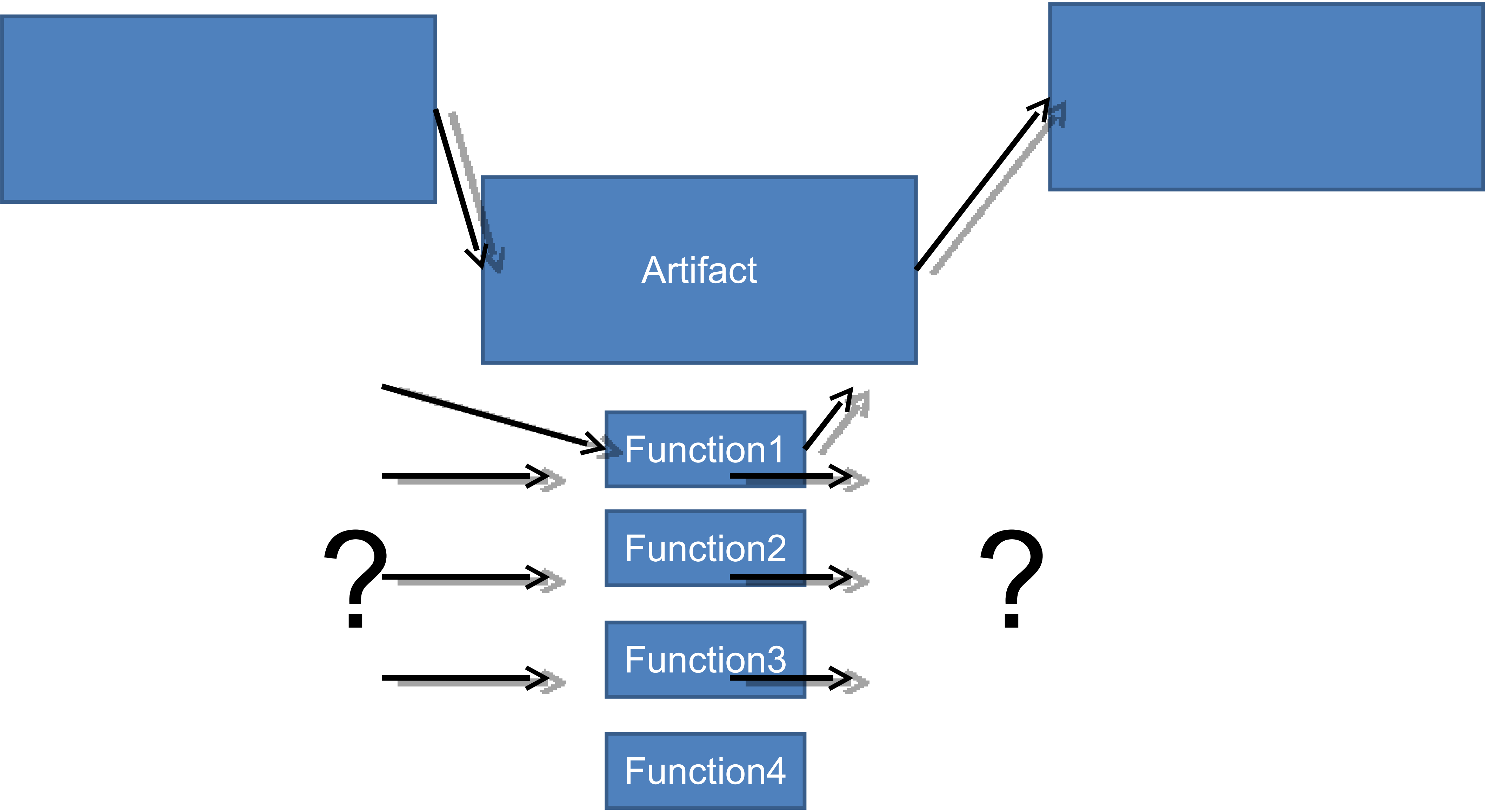
New	Used
0-50 pcs	0-50 pcs
51-100 pcs	51-100 pcs
101+ pcs	101+ pcs

	🇺🇸 Little Plastic Bricks	🇬🇧 Kat's Bits n' kits	🇺🇸 Cincinnati Bricks	🇬🇧 BricksNBitz	🇬🇧 a brick or more	🇫🇷 1001bricks	🇨🇦 Bricks all over	🇺🇸 Daytona Bricks	🇩🇪 MT-Bricks	🇩🇪 Magic Magnus – High-Quality bricks
■ Black	10¢	7¢	6¢	6¢	10¢ 8¢	12¢	7¢ 7¢	7¢ 5¢	6¢	
■ Blue	13¢	9¢	13¢	12¢ 8¢	13¢	18¢	14¢ 9¢	11¢ 6¢		
☑ ChromeSilver						$9.25				
☑ Clear	26¢			19¢		30¢	23¢		16¢	
■ DkRed				15¢			19¢	11¢		
☑ DkStone		8¢	7¢		10¢	10¢	7¢	7¢ 7¢		
■ Green	37¢				34¢	34¢				
■ Lime						31¢				
							10¢			

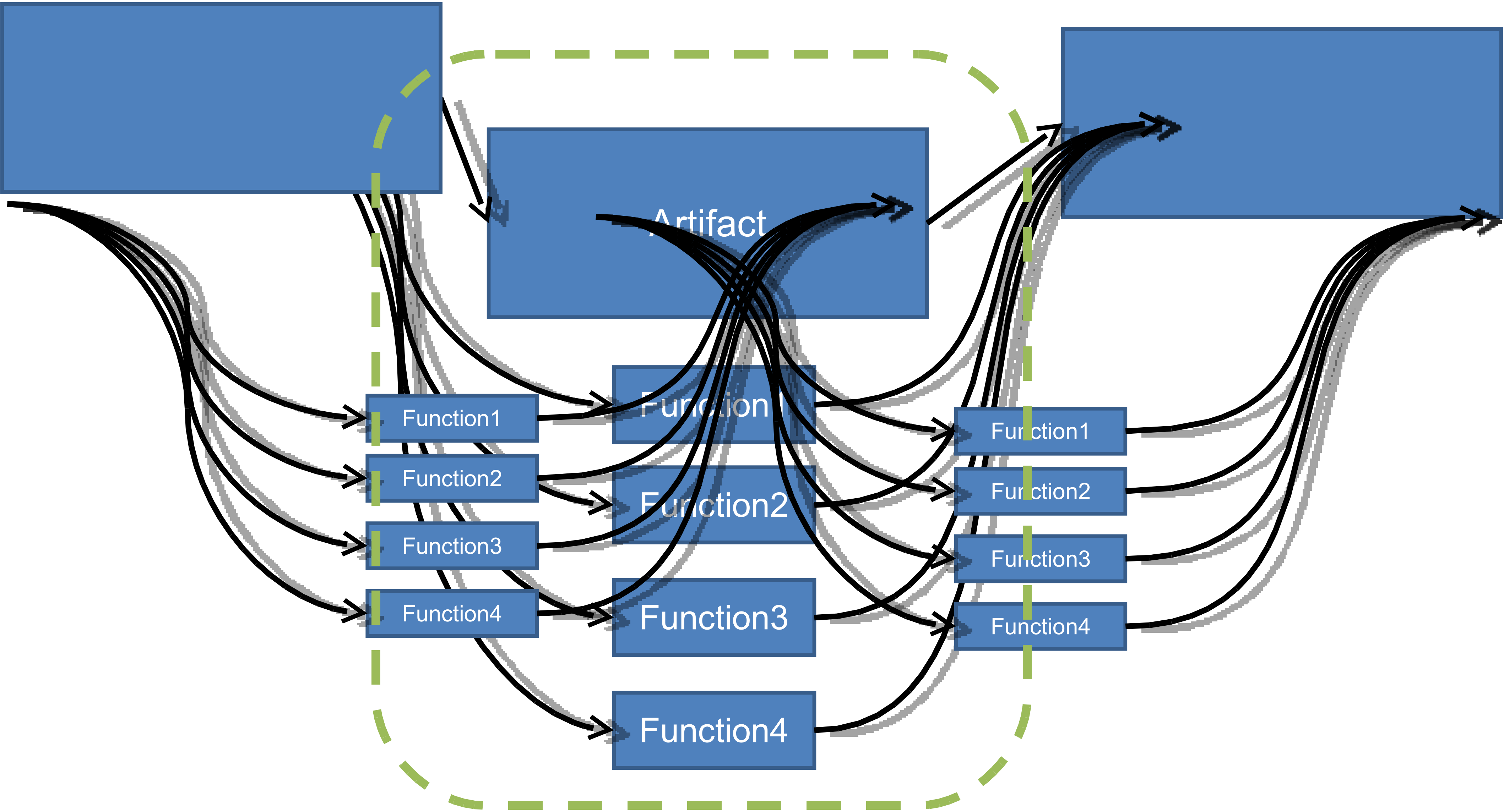
Incompatibilities with UMR repository design

- Hard to contribute to the big locked-in database - single point of failure
- Can't add a new attribute without adding that attribute to every single part in the databse table
- Hard to take the diff of XML files (ordering)
- Flows are confusing (non-quantitative)
- XML schema currently limits possibilities of specification of function (Function Structure Graphs)

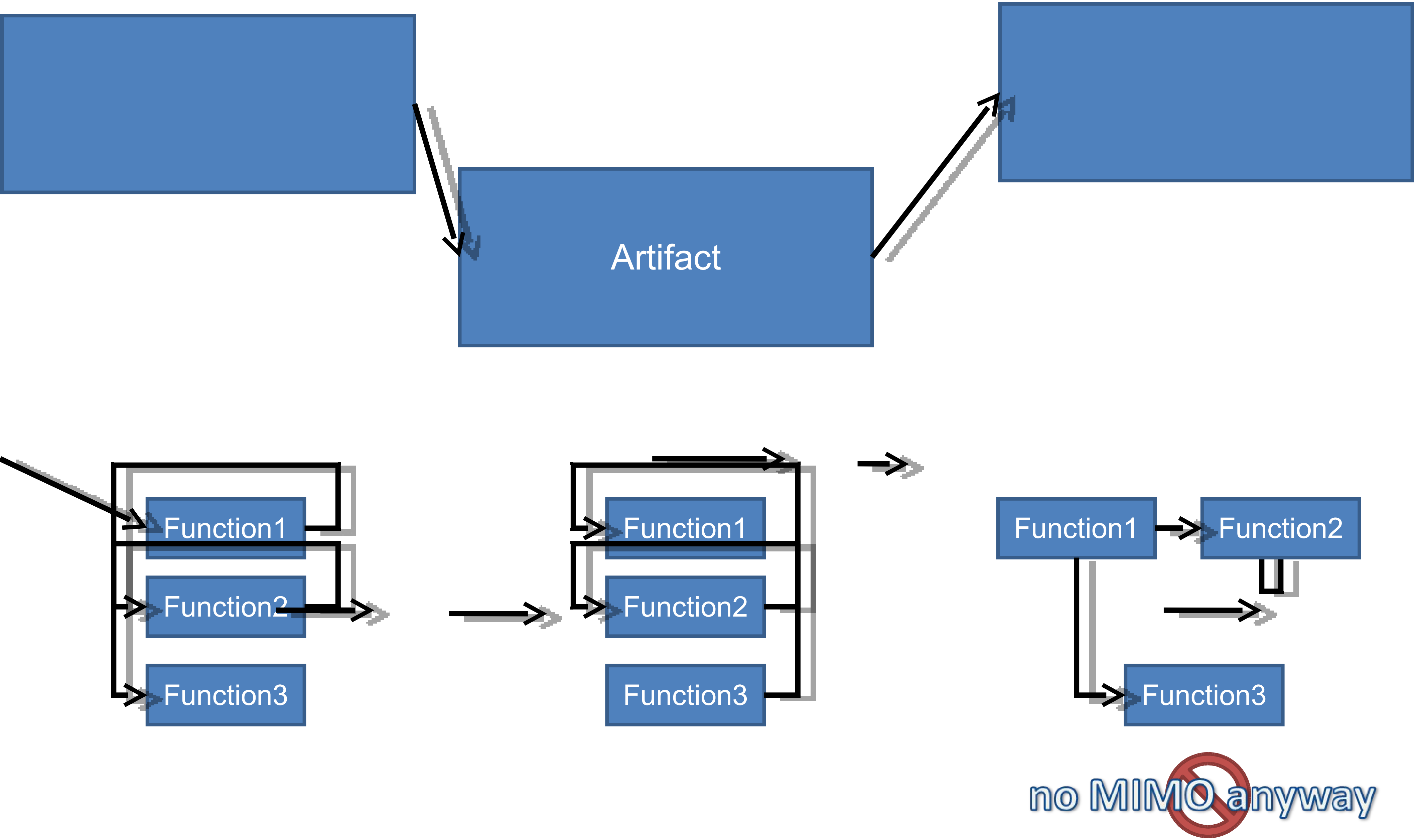
Current Model(?)



Current Model(?)

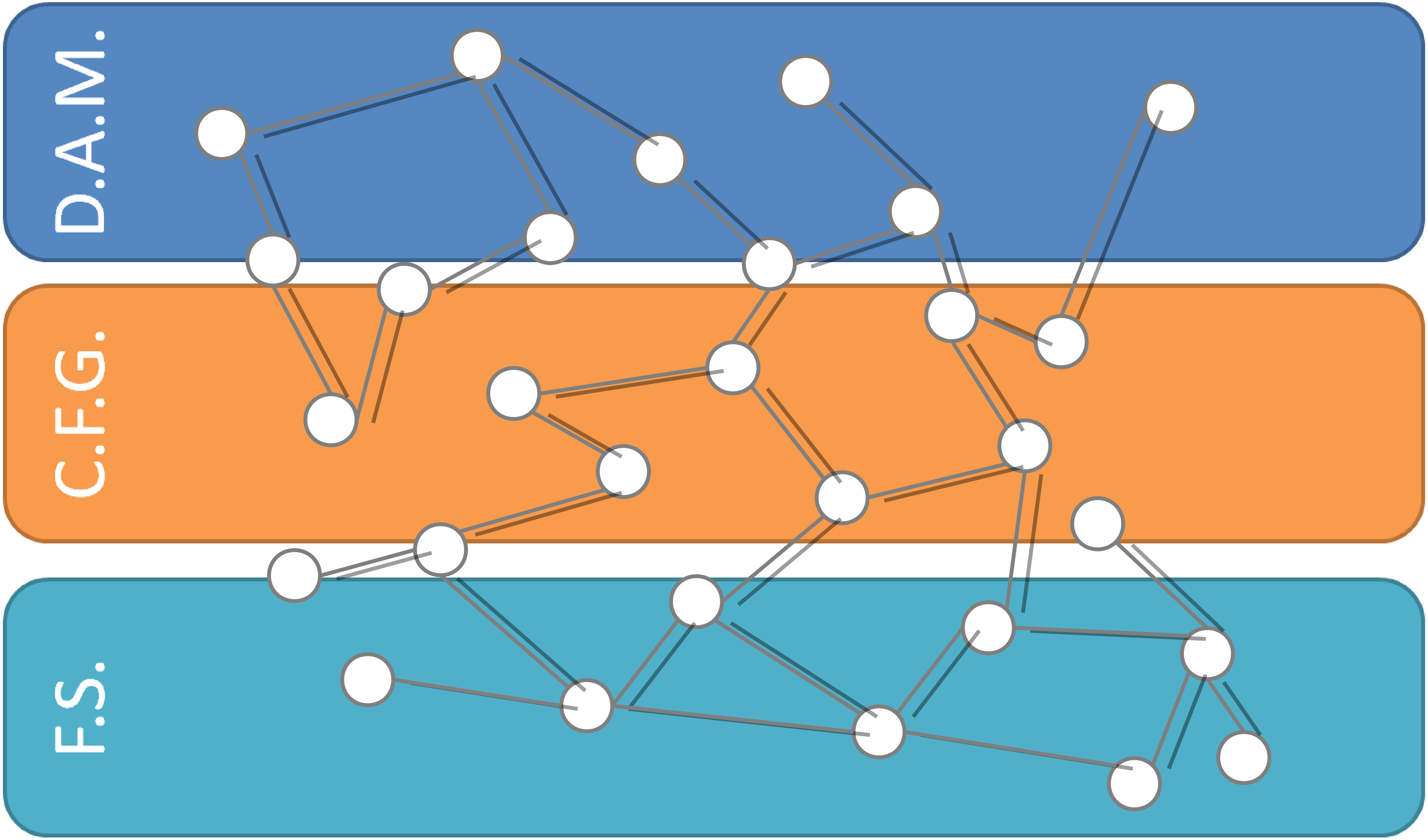


Ambiguity #1: how are functions connected within an artifact?



no MIMO anyway

Consideration: product as a graph



Edges represent interface connections.

Consideration: allow for queries like…

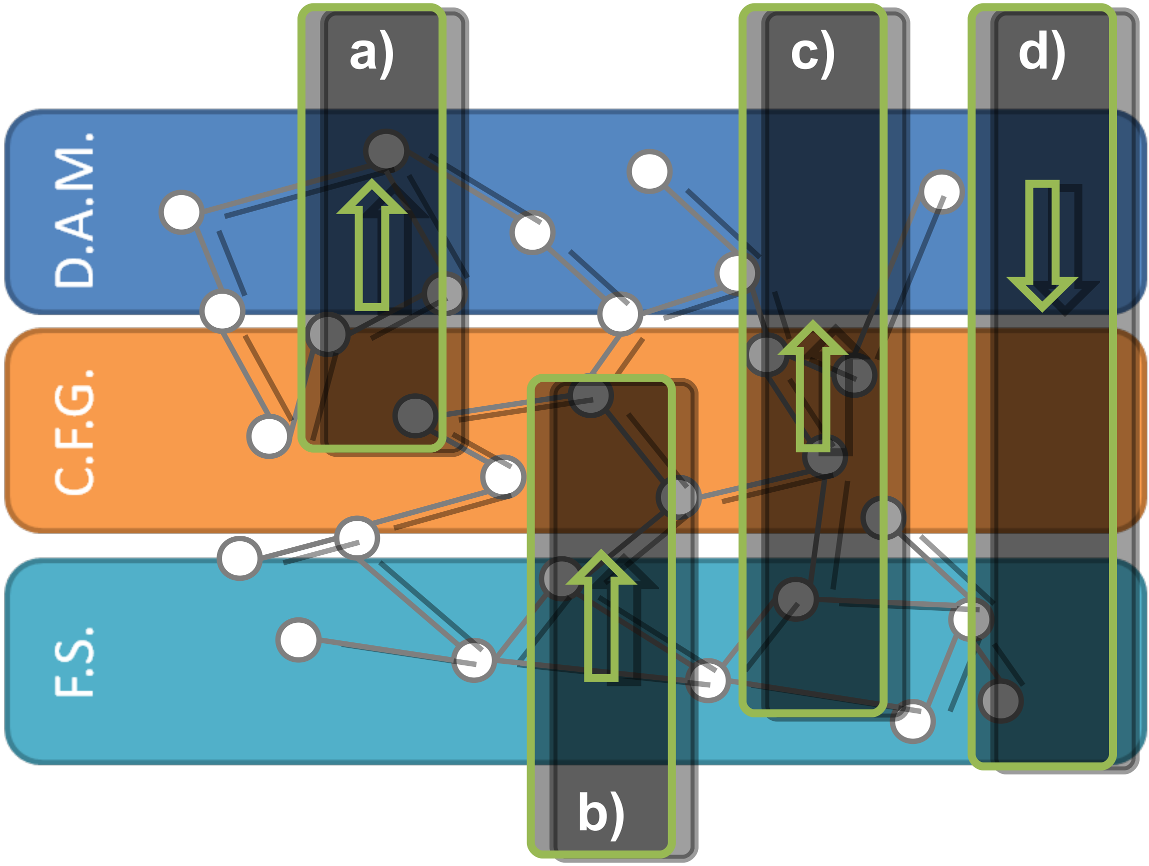
a) How are gear and motor typically held together?

b) How is the function convert fulfilled? Or “Convert EE” or “Convert EE to RME”.

c) Give me a solution for “Guide Solid” from a real product (include connectivity – supporting functions).

d) What does a bolt through a spring do? (What is the function?)

e) Retrieve artifact’s name (gather stats. via FCM) or actual parts used in past simliar design.



a) Within product versus across repository?

Suggestions and Further Collaboration

- Future direction of UMR repository?
- UMR-trained package maintainers can help enable standardization of hardware packages
- Adoptable milestones:
  - Unit tests for entire VOICED / engineering design framework
  - Work out kinks in packaging format and work-flow
  - Can a UMR hardware package interface with a UT package?
  - Algorithms for dependency resolution, instruction generation, Frankenstein concepts
- How can we be of assistance?

Taking a look at skdb

- Repository (git): http://adl.serveftp.org/skdb.git/
- Viewable on the web: http://adl.serveftp.org/git/gitweb.cgi
- Also on github with pretty syntax highlighting:
    - http://github.com/kanzure/skdb
    - http://github.com/kanzure/skdb.git
- Getting assistance
    - IRC: #hplusroadmap on irc.freenode.net
    - Email: openmanufacturing@googlegroups.com
    - Phone: #512-203-0507 (Bryan)
-