

CAD Reusability and the Role of Modeling Information in the MBE Context

Jorge D. Camba

jdorribo@uh.edu

UNIVERSITY of
HOUSTON

Manuel Contero

mcontero@upv.es



Pedro Company

pcompany@uji.es



Agenda

01

Context and Background

- MBE
- Digital Thread

02

Key Terms

- Product Model
- Communication
- Reuse

03

Software Analogy

- Source Code
- Documentation
- Software Reuse

04

Vision

- Quality Modeling
- Quality Testing
- Model Enrichment

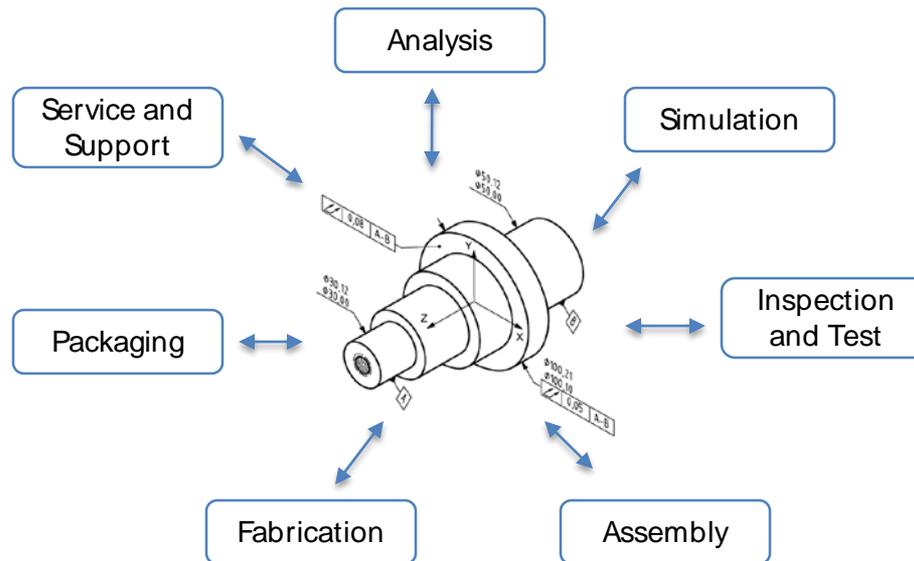
05

Contributions

- Demo

Model-Based Enterprise

- A Model-based Enterprise (MBE) is an organization that applies modeling and simulation technologies to integrate and manage all of its technical and business processes related to production, support, and product retirement¹.
- It applies product and process models at every step of the product life-cycle
- The core MBE tenet is that **data is created once and directly reused** by all data consumers¹.



¹Frechette, S.P. (2011). *Model Based Enterprise for Manufacturing*. 44th CIRP Int. Conf. on Manufacturing Systems. Madison, WI.

Digital Thread

- The Digital Thread for Smart Manufacturing project enables the **repurposing, reuse, and traceability of information throughout the product lifecycle**...Research focuses on standards and implementation needs to exchange information between each phases of the lifecycle - particularly between engineering, manufacturing, and quality functions¹.



¹NIST, Systems Integration Division. *Enabling the Digital Thread for Smart Manufacturing* (<https://go.usa.gov/6nPh>)

“Over-the-wall” Design

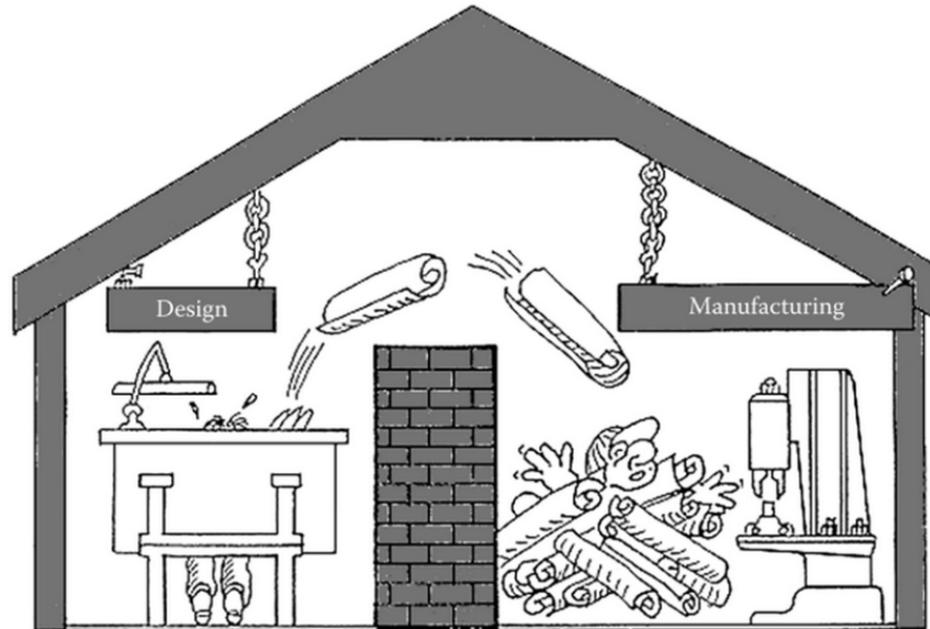
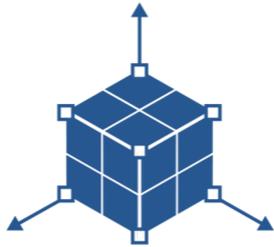


FIGURE 1.6
“Over-the-wall” design, historically the way of doing business.

Source: Boothroyd, G., Dewhurst, P., & Knight, W. A. (2010). *Product design for manufacture and assembly*. CRC press.

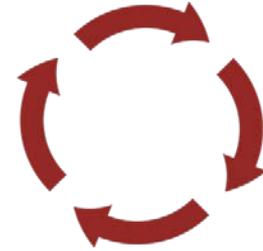
Key Terms



Product Model

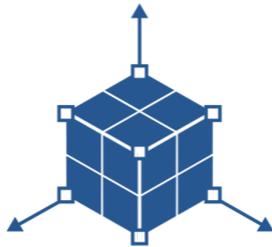


Communication



Reuse

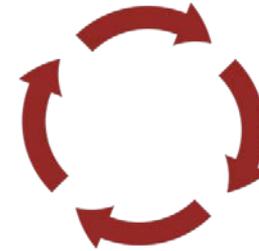
Key Terms



Product Model



Communication

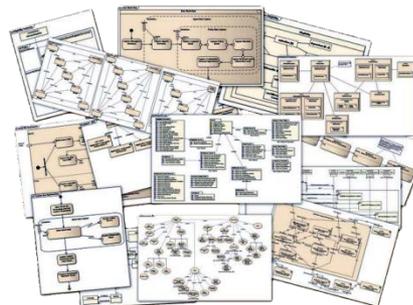


Reuse

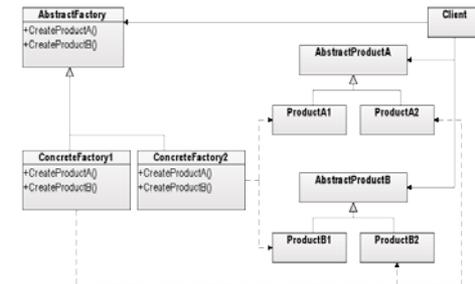
An Analogy: Software Engineering

```
17 string sInput;  
18 int iLength, iN;  
19 double dblTemp;  
20 bool again = true;  
21  
22 while (again) {  
23     iN = -1;  
24     again = false;  
25     getline(cin, sInput);  
26     stringstream(sInput) >> dblTemp;  
27     iLength = sInput.length();  
28     if (iLength < 4) {  
29         again = true;  
30         continue;  
31     } else if (sInput[iLength - 3] != '.') {  
32         again = true;  
33         continue;  
34     } while (++iN < iLength) {  
35         if (isdigit(sInput[iN])) {  
36             continue;  
37         } else if (iN == (iLength - 3)) {  
38             break;  
39         }  
40     }  
41     if (iN < 4) {  
42         again = true;  
43         continue;  
44     }  
45     if (iN > 4) {  
46         again = true;  
47         continue;  
48     }  
49     if (iN < 4) {  
50         again = true;  
51         continue;  
52     }  
53     if (iN > 4) {  
54         again = true;  
55         continue;  
56     }  
57     if (iN < 4) {  
58         again = true;  
59         continue;  
60     }  
61     if (iN > 4) {  
62         again = true;  
63         continue;  
64     }  
65     if (iN < 4) {  
66         again = true;  
67         continue;  
68     }  
69     if (iN > 4) {  
70         again = true;  
71         continue;  
72     }  
73     if (iN < 4) {  
74         again = true;  
75         continue;  
76     }  
77     if (iN > 4) {  
78         again = true;  
79         continue;  
80     }  
81     if (iN < 4) {  
82         again = true;  
83         continue;  
84     }  
85     if (iN > 4) {  
86         again = true;  
87         continue;  
88     }  
89     if (iN < 4) {  
90         again = true;  
91         continue;  
92     }  
93     if (iN > 4) {  
94         again = true;  
95         continue;  
96     }  
97     if (iN < 4) {  
98         again = true;  
99         continue;  
100    }  
101    if (iN > 4) {  
102        again = true;  
103        continue;  
104    }  
105    if (iN < 4) {  
106        again = true;  
107        continue;  
108    }  
109    if (iN > 4) {  
110        again = true;  
111        continue;  
112    }  
113    if (iN < 4) {  
114        again = true;  
115        continue;  
116    }  
117    if (iN > 4) {  
118        again = true;  
119        continue;  
120    }  
121    if (iN < 4) {  
122        again = true;  
123        continue;  
124    }  
125    if (iN > 4) {  
126        again = true;  
127        continue;  
128    }  
129    if (iN < 4) {  
130        again = true;  
131        continue;  
132    }  
133    if (iN > 4) {  
134        again = true;  
135        continue;  
136    }  
137    if (iN < 4) {  
138        again = true;  
139        continue;  
140    }  
141    if (iN > 4) {  
142        again = true;  
143        continue;  
144    }  
145    if (iN < 4) {  
146        again = true;  
147        continue;  
148    }  
149    if (iN > 4) {  
150        again = true;  
151        continue;  
152    }  
153    if (iN < 4) {  
154        again = true;  
155        continue;  
156    }  
157    if (iN > 4) {  
158        again = true;  
159        continue;  
160    }  
161    if (iN < 4) {  
162        again = true;  
163        continue;  
164    }  
165    if (iN > 4) {  
166        again = true;  
167        continue;  
168    }  
169    if (iN < 4) {  
170        again = true;  
171        continue;  
172    }  
173    if (iN > 4) {  
174        again = true;  
175        continue;  
176    }  
177    if (iN < 4) {  
178        again = true;  
179        continue;  
180    }  
181    if (iN > 4) {  
182        again = true;  
183        continue;  
184    }  
185    if (iN < 4) {  
186        again = true;  
187        continue;  
188    }  
189    if (iN > 4) {  
190        again = true;  
191        continue;  
192    }  
193    if (iN < 4) {  
194        again = true;  
195        continue;  
196    }  
197    if (iN > 4) {  
198        again = true;  
199        continue;  
200    }  
201    if (iN < 4) {  
202        again = true;  
203        continue;  
204    }  
205    if (iN > 4) {  
206        again = true;  
207        continue;  
208    }  
209    if (iN < 4) {  
210        again = true;  
211        continue;  
212    }  
213    if (iN > 4) {  
214        again = true;  
215        continue;  
216    }  
217    if (iN < 4) {  
218        again = true;  
219        continue;  
220    }  
221    if (iN > 4) {  
222        again = true;  
223        continue;  
224    }  
225    if (iN < 4) {  
226        again = true;  
227        continue;  
228    }  
229    if (iN > 4) {  
230        again = true;  
231        continue;  
232    }  
233    if (iN < 4) {  
234        again = true;  
235        continue;  
236    }  
237    if (iN > 4) {  
238        again = true;  
239        continue;  
240    }  
241    if (iN < 4) {  
242        again = true;  
243        continue;  
244    }  
245    if (iN > 4) {  
246        again = true;  
247        continue;  
248    }  
249    if (iN < 4) {  
250        again = true;  
251        continue;  
252    }  
253    if (iN > 4) {  
254        again = true;  
255        continue;  
256    }  
257    if (iN < 4) {  
258        again = true;  
259        continue;  
260    }  
261    if (iN > 4) {  
262        again = true;  
263        continue;  
264    }  
265    if (iN < 4) {  
266        again = true;  
267        continue;  
268    }  
269    if (iN > 4) {  
270        again = true;  
271        continue;  
272    }  
273    if (iN < 4) {  
274        again = true;  
275        continue;  
276    }  
277    if (iN > 4) {  
278        again = true;  
279        continue;  
280    }  
281    if (iN < 4) {  
282        again = true;  
283        continue;  
284    }  
285    if (iN > 4) {  
286        again = true;  
287        continue;  
288    }  
289    if (iN < 4) {  
290        again = true;  
291        continue;  
292    }  
293    if (iN > 4) {  
294        again = true;  
295        continue;  
296    }  
297    if (iN < 4) {  
298        again = true;  
299        continue;  
300    }  
301    if (iN > 4) {  
302        again = true;  
303        continue;  
304    }  
305    if (iN < 4) {  
306        again = true;  
307        continue;  
308    }  
309    if (iN > 4) {  
310        again = true;  
311        continue;  
312    }  
313    if (iN < 4) {  
314        again = true;  
315        continue;  
316    }  
317    if (iN > 4) {  
318        again = true;  
319        continue;  
320    }  
321    if (iN < 4) {  
322        again = true;  
323        continue;  
324    }  
325    if (iN > 4) {  
326        again = true;  
327        continue;  
328    }  
329    if (iN < 4) {  
330        again = true;  
331        continue;  
332    }  
333    if (iN > 4) {  
334        again = true;  
335        continue;  
336    }  
337    if (iN < 4) {  
338        again = true;  
339        continue;  
340    }  
341    if (iN > 4) {  
342        again = true;  
343        continue;  
344    }  
345    if (iN < 4) {  
346        again = true;  
347        continue;  
348    }  
349    if (iN > 4) {  
350        again = true;  
351        continue;  
352    }  
353    if (iN < 4) {  
354        again = true;  
355        continue;  
356    }  
357    if (iN > 4) {  
358        again = true;  
359        continue;  
360    }  
361    if (iN < 4) {  
362        again = true;  
363        continue;  
364    }  
365    if (iN > 4) {  
366        again = true;  
367        continue;  
368    }  
369    if (iN < 4) {  
370        again = true;  
371        continue;  
372    }  
373    if (iN > 4) {  
374        again = true;  
375        continue;  
376    }  
377    if (iN < 4) {  
378        again = true;  
379        continue;  
380    }  
381    if (iN > 4) {  
382        again = true;  
383        continue;  
384    }  
385    if (iN < 4) {  
386        again = true;  
387        continue;  
388    }  
389    if (iN > 4) {  
390        again = true;  
391        continue;  
392    }  
393    if (iN < 4) {  
394        again = true;  
395        continue;  
396    }  
397    if (iN > 4) {  
398        again = true;  
399        continue;  
400    }  
401    if (iN < 4) {  
402        again = true;  
403        continue;  
404    }  
405    if (iN > 4) {  
406        again = true;  
407        continue;  
408    }  
409    if (iN < 4) {  
410        again = true;  
411        continue;  
412    }  
413    if (iN > 4) {  
414        again = true;  
415        continue;  
416    }  
417    if (iN < 4) {  
418        again = true;  
419        continue;  
420    }  
421    if (iN > 4) {  
422        again = true;  
423        continue;  
424    }  
425    if (iN < 4) {  
426        again = true;  
427        continue;  
428    }  
429    if (iN > 4) {  
430        again = true;  
431        continue;  
432    }  
433    if (iN < 4) {  
434        again = true;  
435        continue;  
436    }  
437    if (iN > 4) {  
438        again = true;  
439        continue;  
440    }  
441    if (iN < 4) {  
442        again = true;  
443        continue;  
444    }  
445    if (iN > 4) {  
446        again = true;  
447        continue;  
448    }  
449    if (iN < 4) {  
450        again = true;  
451        continue;  
452    }  
453    if (iN > 4) {  
454        again = true;  
455        continue;  
456    }  
457    if (iN < 4) {  
458        again = true;  
459        continue;  
460    }  
461    if (iN > 4) {  
462        again = true;  
463        continue;  
464    }  
465    if (iN < 4) {  
466        again = true;  
467        continue;  
468    }  
469    if (iN > 4) {  
470        again = true;  
471        continue;  
472    }  
473    if (iN < 4) {  
474        again = true;  
475        continue;  
476    }  
477    if (iN > 4) {  
478        again = true;  
479        continue;  
480    }  
481    if (iN < 4) {  
482        again = true;  
483        continue;  
484    }  
485    if (iN > 4) {  
486        again = true;  
487        continue;  
488    }  
489    if (iN < 4) {  
490        again = true;  
491        continue;  
492    }  
493    if (iN > 4) {  
494        again = true;  
495        continue;  
496    }  
497    if (iN < 4) {  
498        again = true;  
499        continue;  
500    }  
501    if (iN > 4) {  
502        again = true;  
503        continue;  
504    }  
505    if (iN < 4) {  
506        again = true;  
507        continue;  
508    }  
509    if (iN > 4) {  
510        again = true;  
511        continue;  
512    }  
513    if (iN < 4) {  
514        again = true;  
515        continue;  
516    }  
517    if (iN > 4) {  
518        again = true;  
519        continue;  
520    }  
521    if (iN < 4) {  
522        again = true;  
523        continue;  
524    }  
525    if (iN > 4) {  
526        again = true;  
527        continue;  
528    }  
529    if (iN < 4) {  
530        again = true;  
531        continue;  
532    }  
533    if (iN > 4) {  
534        again = true;  
535        continue;  
536    }  
537    if (iN < 4) {  
538        again = true;  
539        continue;  
540    }  
541    if (iN > 4) {  
542        again = true;  
543        continue;  
544    }  
545    if (iN < 4) {  
546        again = true;  
547        continue;  
548    }  
549    if (iN > 4) {  
550        again = true;  
551        continue;  
552    }  
553    if (iN < 4) {  
554        again = true;  
555        continue;  
556    }  
557    if (iN > 4) {  
558        again = true;  
559        continue;  
560    }  
561    if (iN < 4) {  
562        again = true;  
563        continue;  
564    }  
565    if (iN > 4) {  
566        again = true;  
567        continue;  
568    }  
569    if (iN < 4) {  
570        again = true;  
571        continue;  
572    }  
573    if (iN > 4) {  
574        again = true;  
575        continue;  
576    }  
577    if (iN < 4) {  
578        again = true;  
579        continue;  
580    }  
581    if (iN > 4) {  
582        again = true;  
583        continue;  
584    }  
585    if (iN < 4) {  
586        again = true;  
587        continue;  
588    }  
589    if (iN > 4) {  
590        again = true;  
591        continue;  
592    }  
593    if (iN < 4) {  
594        again = true;  
595        continue;  
596    }  
597    if (iN > 4) {  
598        again = true;  
599        continue;  
600    }  
601    if (iN < 4) {  
602        again = true;  
603        continue;  
604    }  
605    if (iN > 4) {  
606        again = true;  
607        continue;  
608    }  
609    if (iN < 4) {  
610        again = true;  
611        continue;  
612    }  
613    if (iN > 4) {  
614        again = true;  
615        continue;  
616    }  
617    if (iN < 4) {  
618        again = true;  
619        continue;  
620    }  
621    if (iN > 4) {  
622        again = true;  
623        continue;  
624    }  
625    if (iN < 4) {  
626        again = true;  
627        continue;  
628    }  
629    if (iN > 4) {  
630        again = true;  
631        continue;  
632    }  
633    if (iN < 4) {  
634        again = true;  
635        continue;  
636    }  
637    if (iN > 4) {  
638        again = true;  
639        continue;  
640    }  
641    if (iN < 4) {  
642        again = true;  
643        continue;  
644    }  
645    if (iN > 4) {  
646        again = true;  
647        continue;  
648    }  
649    if (iN < 4) {  
650        again = true;  
651        continue;  
652    }  
653    if (iN > 4) {  
654        again = true;  
655        continue;  
656    }  
657    if (iN < 4) {  
658        again = true;  
659        continue;  
660    }  
661    if (iN > 4) {  
662        again = true;  
663        continue;  
664    }  
665    if (iN < 4) {  
666        again = true;  
667        continue;  
668    }  
669    if (iN > 4) {  
670        again = true;  
671        continue;  
672    }  
673    if (iN < 4) {  
674        again = true;  
675        continue;  
676    }  
677    if (iN > 4) {  
678        again = true;  
679        continue;  
680    }  
681    if (iN < 4) {  
682        again = true;  
683        continue;  
684    }  
685    if (iN > 4) {  
686        again = true;  
687        continue;  
688    }  
689    if (iN < 4) {  
690        again = true;  
691        continue;  
692    }  
693    if (iN > 4) {  
694        again = true;  
695        continue;  
696    }  
697    if (iN < 4) {  
698        again = true;  
699        continue;  
700    }  
701    if (iN > 4) {  
702        again = true;  
703        continue;  
704    }  
705    if (iN < 4) {  
706        again = true;  
707        continue;  
708    }  
709    if (iN > 4) {  
710        again = true;  
711        continue;  
712    }  
713    if (iN < 4) {  
714        again = true;  
715        continue;  
716    }  
717    if (iN > 4) {  
718        again = true;  
719        continue;  
720    }  
721    if (iN < 4) {  
722        again = true;  
723        continue;  
724    }  
725    if (iN > 4) {  
726        again = true;  
727        continue;  
728    }  
729    if (iN < 4) {  
730        again = true;  
731        continue;  
732    }  
733    if (iN > 4) {  
734        again = true;  
735        continue;  
736    }  
737    if (iN < 4) {  
738        again = true;  
739        continue;  
740    }  
741    if (iN > 4) {  
742        again = true;  
743        continue;  
744    }  
745    if (iN < 4) {  
746        again = true;  
747        continue;  
748    }  
749    if (iN > 4) {  
750        again = true;  
751        continue;  
752    }  
753    if (iN < 4) {  
754        again = true;  
755        continue;  
756    }  
757    if (iN > 4) {  
758        again = true;  
759        continue;  
760    }  
761    if (iN < 4) {  
762        again = true;  
763        continue;  
764    }  
765    if (iN > 4) {  
766        again = true;  
767        continue;  
768    }  
769    if (iN < 4) {  
770        again = true;  
771        continue;  
772    }  
773    if (iN > 4) {  
774        again = true;  
775        continue;  
776    }  
777    if (iN < 4) {  
778        again = true;  
779        continue;  
780    }  
781    if (iN > 4) {  
782        again = true;  
783        continue;  
784    }  
785    if (iN < 4) {  
786        again = true;  
787        continue;  
788    }  
789    if (iN > 4) {  
790        again = true;  
791        continue;  
792    }  
793    if (iN < 4) {  
794        again = true;  
795        continue;  
796    }  
797    if (iN > 4) {  
798        again = true;  
799        continue;  
800    }  
801    if (iN < 4) {  
802        again = true;  
803        continue;  
804    }  
805    if (iN > 4) {  
806        again = true;  
807        continue;  
808    }  
809    if (iN < 4) {  
810        again = true;  
811        continue;  
812    }  
813    if (iN > 4) {  
814        again = true;  
815        continue;  
816    }  
817    if (iN < 4) {  
818        again = true;  
819        continue;  
820    }  
821    if (iN > 4) {  
822        again = true;  
823        continue;  
824    }  
825    if (iN < 4) {  
826        again = true;  
827        continue;  
828    }  
829    if (iN > 4) {  
830        again = true;  
831        continue;  
832    }  
833    if (iN < 4) {  
834        again = true;  
835        continue;  
836    }  
837    if (iN > 4) {  
838        again = true;  
839        continue;  
840    }  
841    if (iN < 4) {  
842        again = true;  
843        continue;  
844    }  
845    if (iN > 4) {  
846        again = true;  
847        continue;  
848    }  
849    if (iN < 4) {  
850        again = true;  
851        continue;  
852    }  
853    if (iN > 4) {  
854        again = true;  
855        continue;  
856    }  
857    if (iN < 4) {  
858        again = true;  
859        continue;  
860    }  
861    if (iN > 4) {  
862        again = true;  
863        continue;  
864    }  
865    if (iN < 4) {  
866        again = true;  
867        continue;  
868    }  
869    if (iN > 4) {  
870        again = true;  
871        continue;  
872    }  
873    if (iN < 4) {  
874        again = true;  
875        continue;  
876    }  
877    if (iN > 4) {  
878        again = true;  
879        continue;  
880    }  
881    if (iN < 4) {  
882        again = true;  
883        continue;  
884    }  
885    if (iN > 4) {  
886        again = true;  
887        continue;  
888    }  
889    if (iN < 4) {  
890        again = true;  
891        continue;  
892    }  
893    if (iN > 4) {  
894        again = true;  
895        continue;  
896    }  
897    if (iN < 4) {  
898        again = true;  
899        continue;  
900    }  
901    if (iN > 4) {  
902        again = true;  
903        continue;  
904    }  
905    if (iN < 4) {  
906        again = true;  
907        continue;  
908    }  
909    if (iN > 4) {  
910        again = true;  
911        continue;  
912    }  
913    if (iN < 4) {  
914        again = true;  
915        continue;  
916    }  
917    if (iN > 4) {  
918        again = true;  
919        continue;  
920    }  
921    if (iN < 4) {  
922        again = true;  
923        continue;  
924    }  
925    if (iN > 4) {  
926        again = true;  
927        continue;  
928    }  
929    if (iN < 4) {  
930        again = true;  
931        continue;  
932    }  
933    if (iN > 4) {  
934        again = true;  
935        continue;  
936    }  
937    if (iN < 4) {  
938        again = true;  
939        continue;  
940    }  
941    if (iN > 4) {  
942        again = true;  
943        continue;  
944    }  
945    if (iN < 4) {  
946        again = true;  
947        continue;  
948    }  
949    if (iN > 4) {  
950        again = true;  
951        continue;  
952    }  
953    if (iN < 4) {  
954        again = true;  
955        continue;  
956    }  
957    if (iN > 4) {  
958        again = true;  
959        continue;  
960    }  
961    if (iN < 4) {  
962        again = true;  
963        continue;  
964    }  
965    if (iN > 4) {  
966        again = true;  
967        continue;  
968    }  
969    if (iN < 4) {  
970        again = true;  
971        continue;  
972    }  
973    if (iN > 4) {  
974        again = true;  
975        continue;  
976    }  
977    if (iN < 4) {  
978        again = true;  
979        continue;  
980    }  
981    if (iN > 4) {  
982        again = true;  
983        continue;  
984    }  
985    if (iN < 4) {  
986        again = true;  
987        continue;  
988    }  
989    if (iN > 4) {  
990        again = true;  
991        continue;  
992    }  
993    if (iN < 4) {  
994        again = true;  
995        continue;  
996    }  
997    if (iN > 4) {  
998        again = true;  
999        continue;  
1000   }  
1001   }
```

Source Code



Documentation



Software Reuse

An Analogy: Software Engineering

Software Engineering Sub-Disciplines

- Software requirements
- Software design
- Software construction
- Software testing
- Software maintenance
- Software configuration management
- Software engineering management
- Software development process
- Software engineering models and methods
- Software quality
- Software engineering professional
- Software engineering economics
- Computing foundations
- Mathematical foundations
- Engineering foundations

An Analogy: Software Engineering

Software Quality

- The way a program is written can have important consequences.
- Software quality refers to:
 - Functional quality
 - Structural quality
- International standard for the evaluation of software quality: ISO/IEC 25010:2011
 - Functionality, Reliability, Usability , Efficiency, Maintainability, and Portability
- Examples of Strategies:
 - Coding conventions
 - Code refactoring
 - Programming Paradigms (Object-Oriented, Structured, etc.).



Product Model

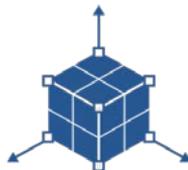
Product Model

- Just like the quality of a software system depends on the correctness and efficiency of its code, the quality of products depends on the quality of their design processes, which then depend on the quality of their data.
- **Product Data Quality (PDQ)** is a measurement of the accuracy and appropriateness of all data involved in the design and manufacturing of a product combined with the timeliness with which they are provided to the stakeholders who may need them^{1,2}.
- Current Model Quality Tools (MQT) are primarily aimed at preventing “easily” solvable low semantic level mistakes and incoherencies³.

¹Yang, J., et al. (2006) *Product data quality assurance for e-manufacturing in the automotive industry*. Int. Journal of Computer Integrated Manufacturing; 19(2): 136-147.

²Strategic Automotive product data Standards Industry Group, SASIG, Product Data Quality Workgroup (PDQ) (2005). *SASIG– Product Data Quality Guidelines for the Global Automotive Industry, Guideline version 2.1* (STEP Part 59. ISO/PAS 26183:2006.)

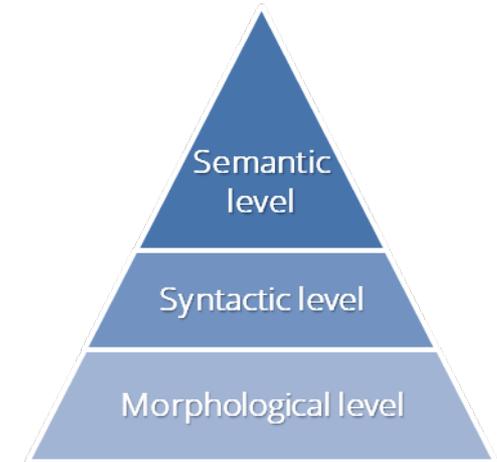
³González-Lluch, C., Company, P., Contero, M., Camba, J D., & Plumed, R. (2017). *A survey on 3D CAD model quality assurance and testing tools*. Computer-Aided Design, 83, 64-79.



Product Model

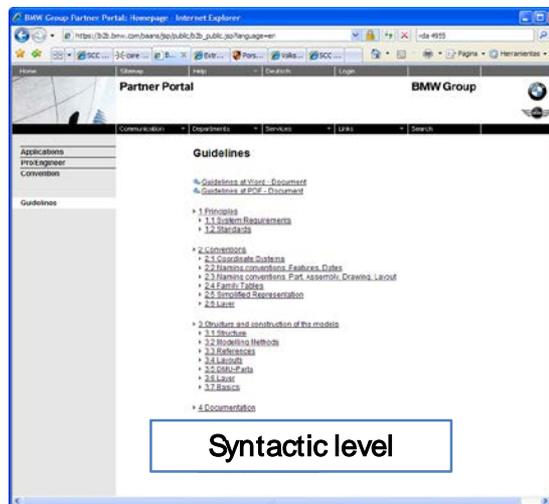
Product Data Quality Model¹

- **Morphological quality:** related to the geometrical and topological correctness of the CAD model
- **Syntactic quality:** evaluates the use of proper modeling conventions.
- **Semantic/pragmatic quality:** takes into account the model's capability for reuse and modification

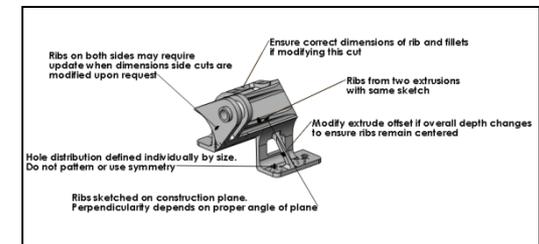


VDA-Recommendation 4955/2		
VDA VDMA	"CAD/CAM" Working Group in the VDA-Raw Material Committee (VDA-AK "CAD/CAM")	4955
Scope and Quality of CAD/CAM data		
This recommendation serves the purpose of defining fundamental, common requirements on the quality, the scope and the control of CAD data.		
This recommendation supplements the VDA recommendation 4950 "Exchange of CAD/CAM data".		
It represents the project results of the VDA working group "Scope and quality of CAD/CAM data", as well as of the VDMA working group "Communication and quality of design data".		
Edition 1	October 1993	
Edition 2	September 1999	

Morphological level



Syntactic level



Semantic level



Product Model

¹Contero, M., Company, P., Vila, C., & Aleixos, N. (2002). *Product data quality and collaborative engineering*. IEEE Computer Graphics and Applications, 22(3), 32-42.

An Analogy: Software Engineering

Communication

- In software engineering, developers use source code comments to support program maintenance and future modifications.

Source code example from Wikipedia

```
/* loop backwards through all elements returned by the  
server (they should be processed chronologically)*/
```

```
for (i = (numElementsReturned - 1); i >= 0; i--)  
{  
    /* process each element's data */  
    updatePattern(i, returnedElements[i]);  
}
```

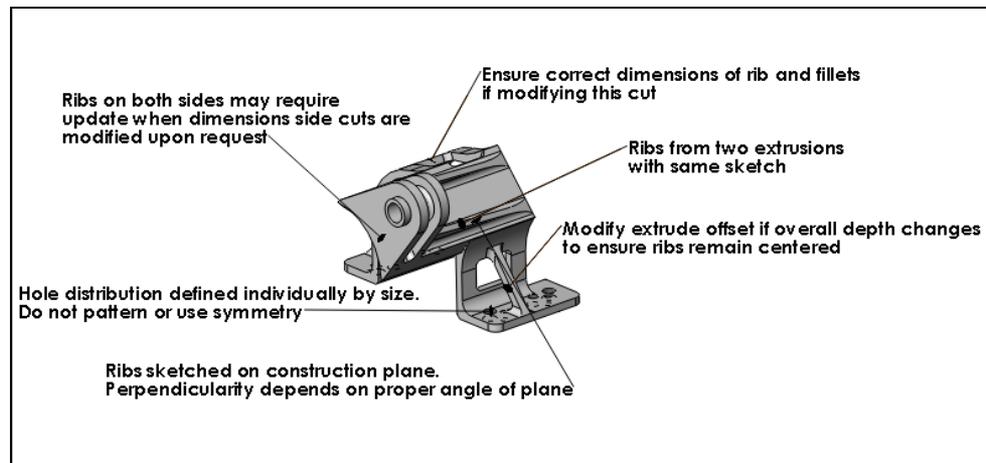
- Tools such as **Doxygen**, **Javadoc**, **Apple's HeaderDoc**, or **Microsoft's Sandcastle** can automatically generate project documentation from source code comments.



Communication

Communication

- Product and Manufacturing Information (PMI)
 - Geometric Dimensioning and Tolerancing (GD&T)
 - 3D annotations (text)
 - Surface finish
 - Material specifications
- Standards for 3D part definition:
 - ASME Y14.41
 - ISO 16792

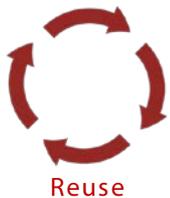


Communication

An Analogy: Software Engineering

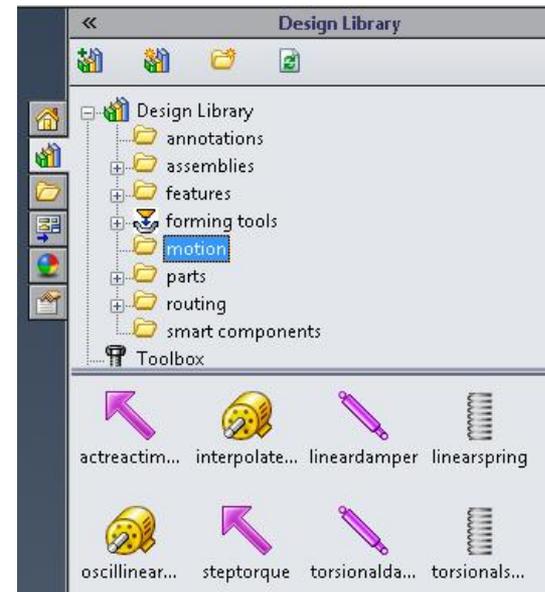
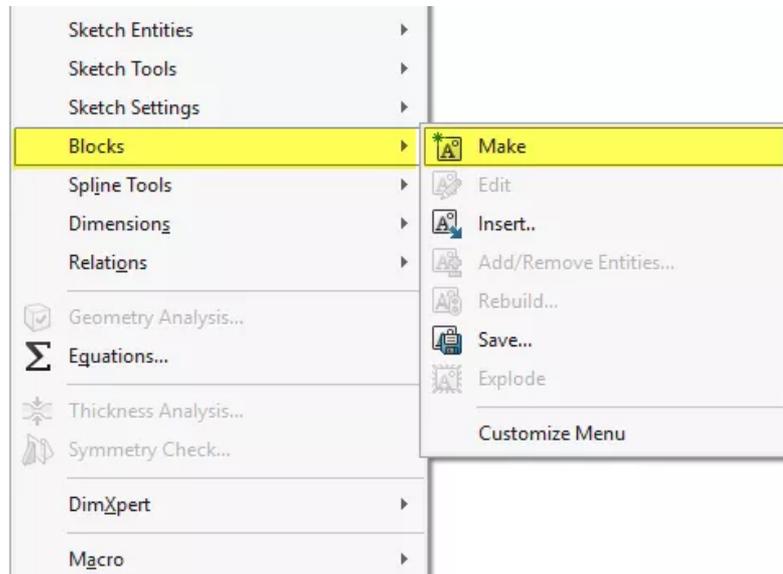
Software Reusability

- “Code reuse is the Holy Grail of Software Engineering.”
 - Douglas Crockford (Senior JavaScript Architect at PayPal)
- Reusability is the use of existing assets in some form within the software product development process, including code, software knowledge, test suites, designs and documentation.
- Examples of Strategies:
 - Subroutines
 - Software Libraries and Components
 - Design Patterns



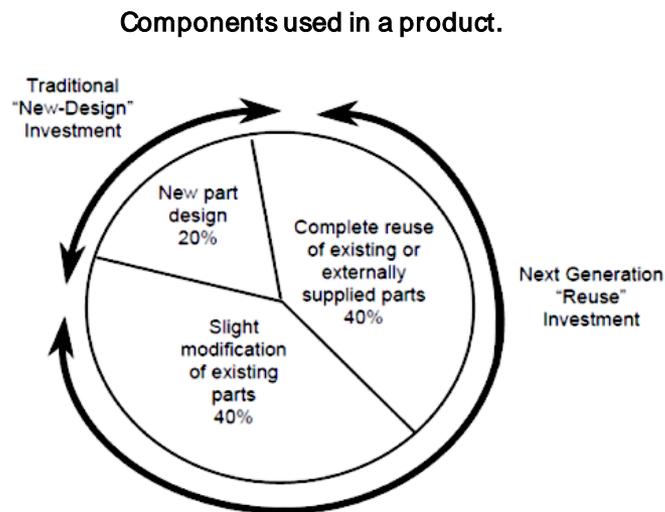
Design Reuse

- Standard Parts and Libraries



Design Reuse

- All engineering organizations report that they reuse designs, but the top performing ones deploy techniques and technologies to CAPITALIZE on design reuse.
- Design Reuse → Model Reuse



Current Use and Future Plans for Reuse Initiatives

Reuse Initiatives	Currently Use	Plan to Use
Reuse of product designs	80%	20%
Reuse of simulation / analysis content	35%	24%
Reuse of NC programming content	29%	23%

Aberdeen Group, *The Design Reuse Benchmark Report* (2007)

Rezayat, M. (2000). *Knowledge-based product development using XML and KCs*. *Computer-aided design*, 32(5), 299-309.



Design Reuse

- There are important obstacles to implement CAD model reuse

Challenges	
Model modification required expert CAD knowledge	57%
Models are inflexible and fail after changes	48%
Only original designer can change models successfully	40%

Aberdeen Group. *The Design Reuse Benchmark Report (2007)*



Design Reuse

- There are important obstacles to implement CAD model reuse

Challenges	
Model modification required expert CAD knowledge	57%
Models are inflexible and fail after changes	48%
Only original designer can change models successfully	40%

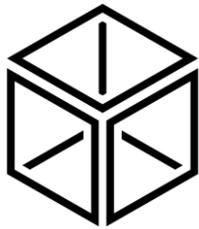
Responses	
Train users to increase CAD skills	71%
Design for wide range of modifications	64%
Detail design information in model (PMI)	36%

Aberdeen Group. *The Design Reuse Benchmark Report (2007)*

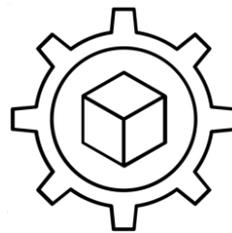


Vision

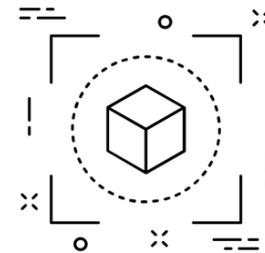
- Improving digital **product representation** and **data quality** and its connection to **design knowledge** at the early stages of the product life-cycle can reduce the complexity of the “digital thread” and lead to more efficient product development processes.
- The execution of this vision is based on three primary ideas:



High Quality Modeling



Model Quality Testing

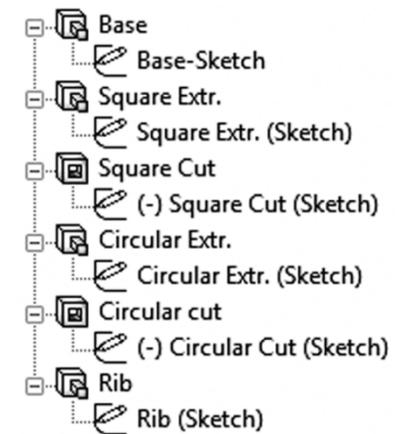
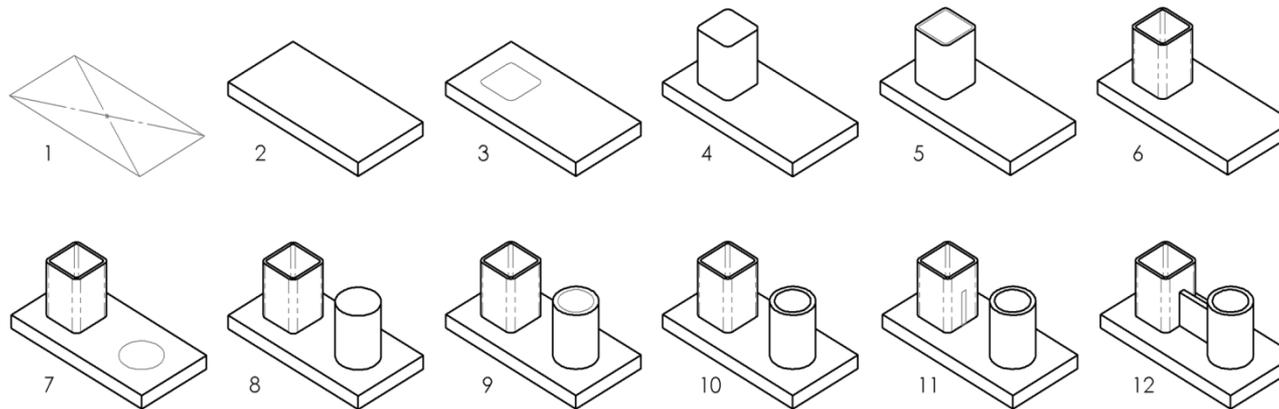


Model Enrichment

1. High Quality Modeling



- History-based parametric CAD modeling

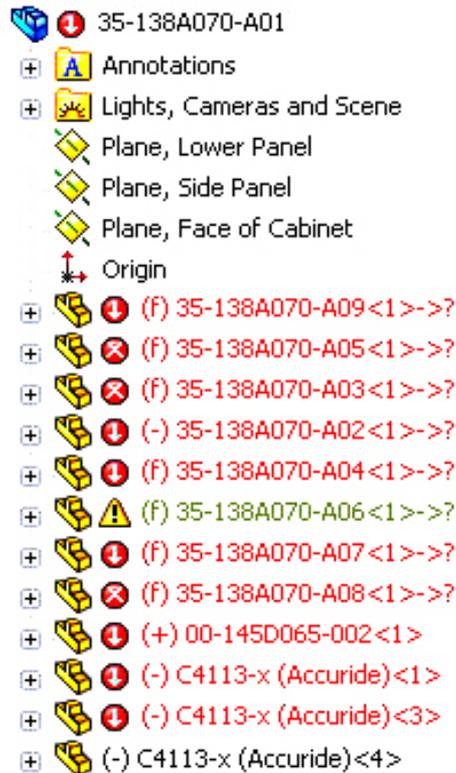


Design Tree
(SolidWorks)

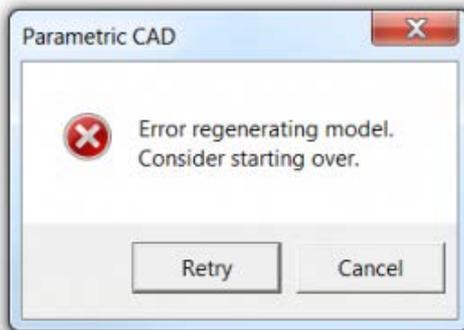
1. High Quality Modeling



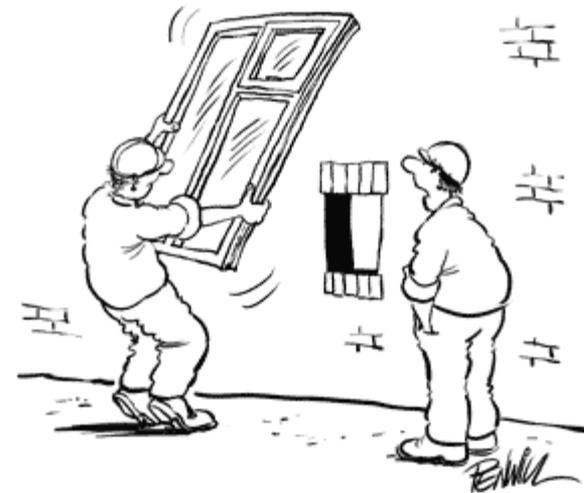
- Rebuild Errors



1. High Quality Modeling



Source: Evan Yares, *The failed promise of parametric CAD*
(<http://www.3dcadworld.com/the-failed-promise-of-parametric-cad/>)



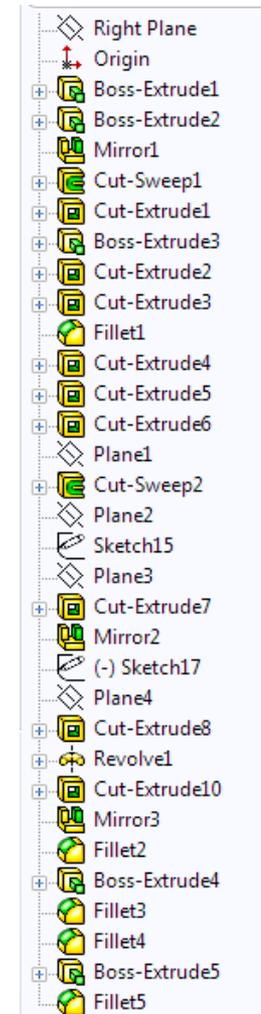
"LOOKS LIKE A PARAMETRIC DATA ERROR
IF YOU ASK ME"

© Roger Penwill



Design Intent

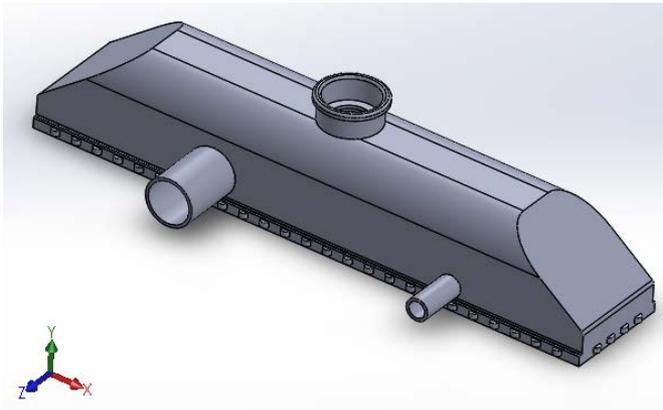
- In parametric CAD, design intent is expressed implicitly in the model's design tree.
- The size and underlying complexity of the design tree can grow rapidly, even for relatively simple models.
- Modeling decisions are not explained.



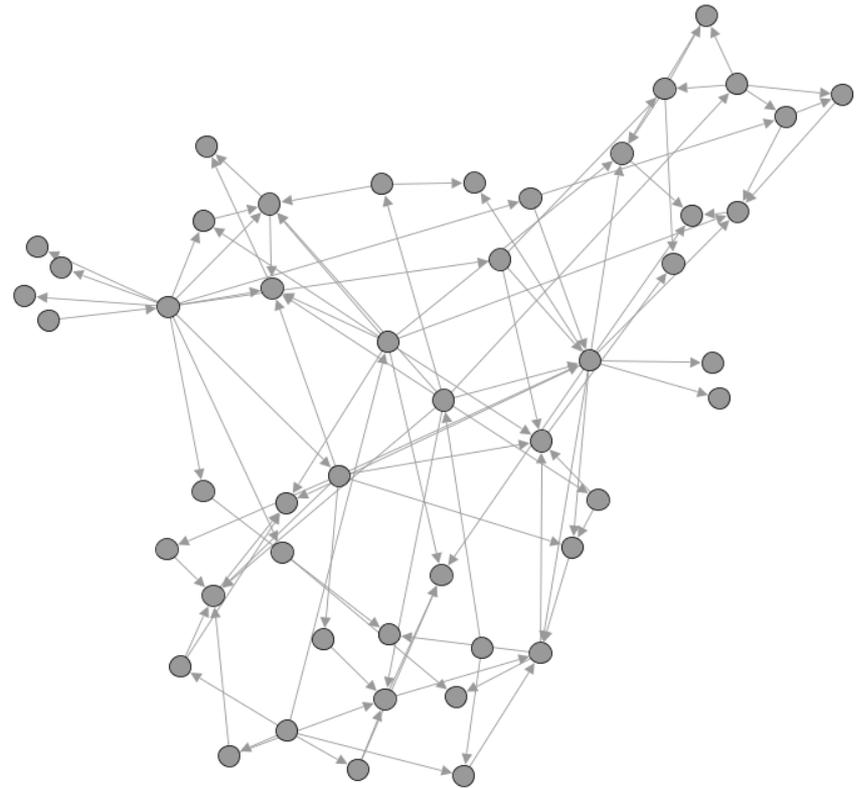
1. High Quality Modeling



- History-based parametric CAD modeling (behind the scenes)



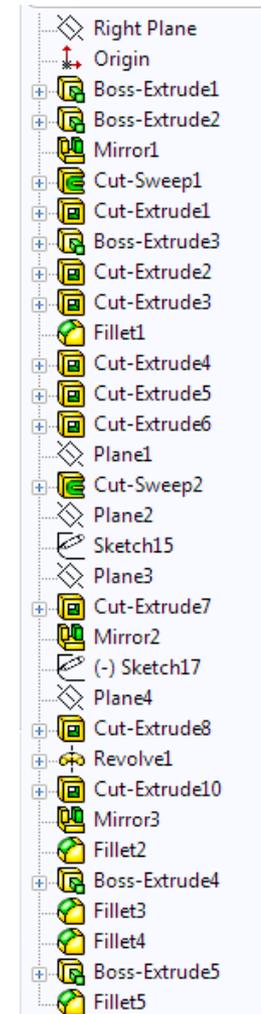
Total Features: 48
Total Dependencies: 97



1. High Quality Modeling



- Parent/child relationships can be problematic.
- The interdependencies of the model should be defined according to the design intent.
- But the design tree is only a sequential list of features...

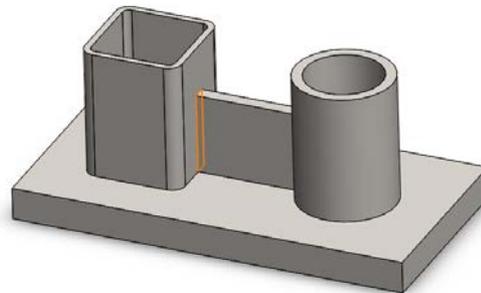
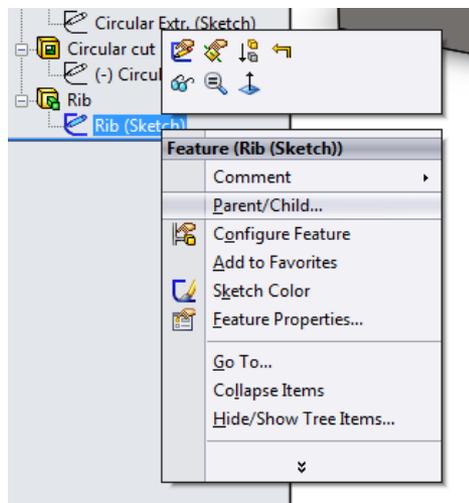


1. High Quality Modeling



Strategies

- Improving the understanding of parent-child relationships



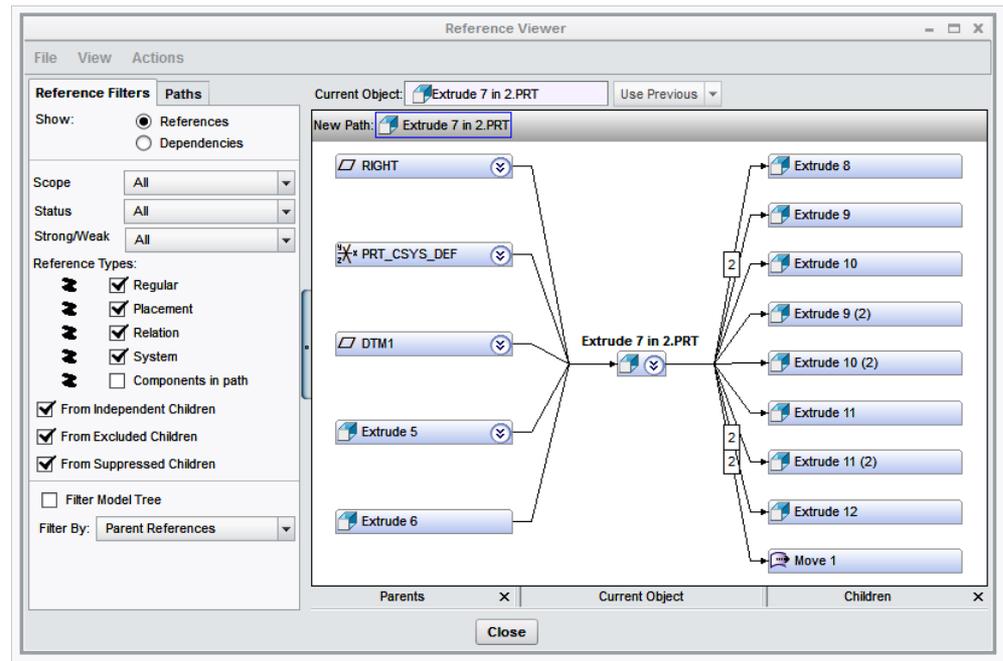
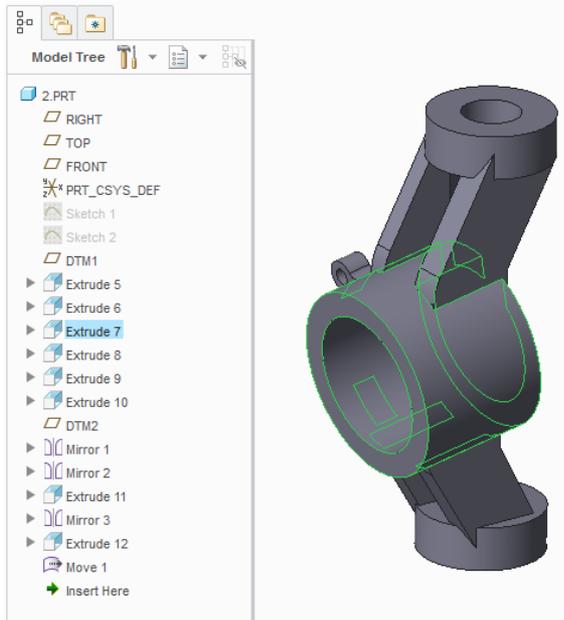
Solidworks

1. High Quality Modeling



Strategies

- Improving the understanding of parent-child relationships



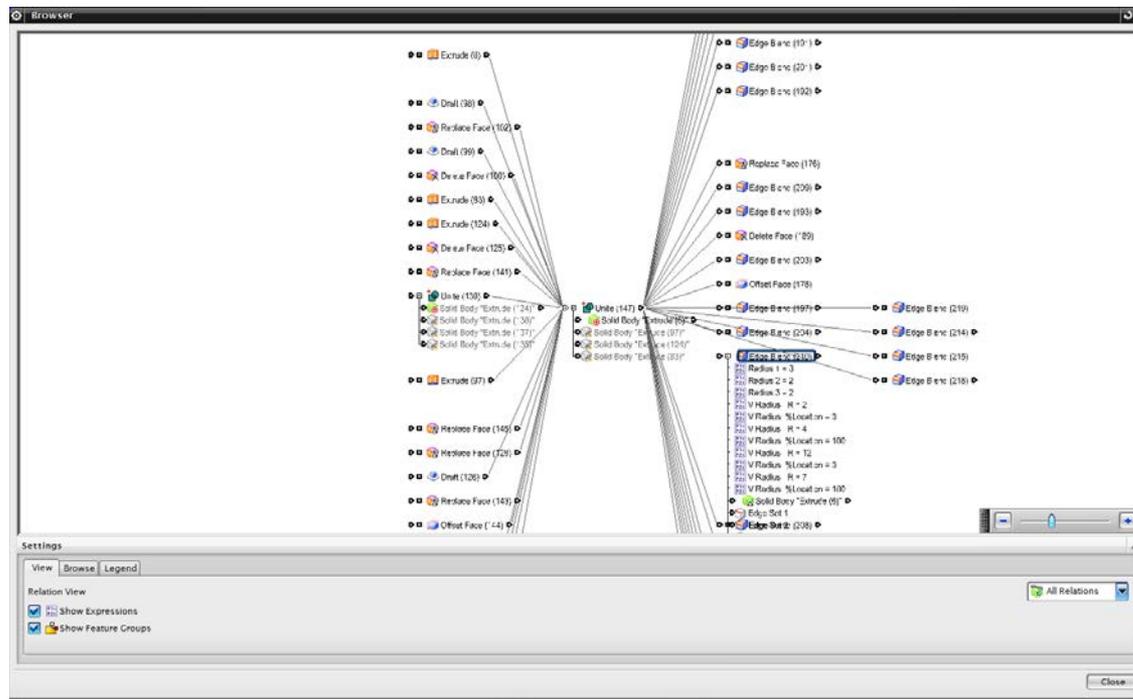
PTC Creo

1. High Quality Modeling



Strategies

- Improving the understanding of parent-child relationships



Parameter	Value	Expression

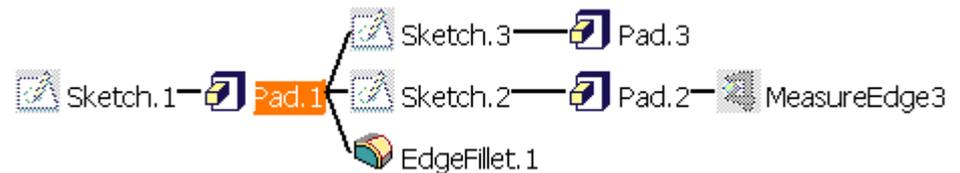
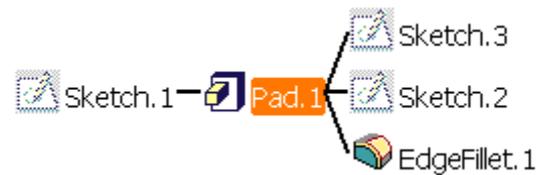
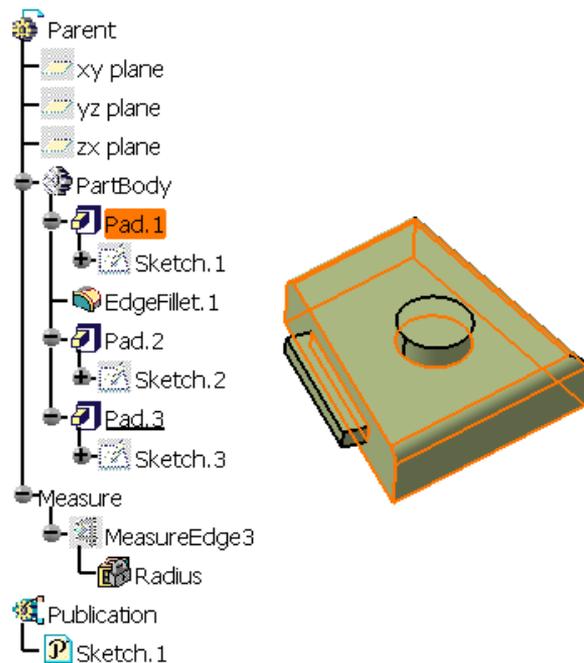
Siemens NX

1. High Quality Modeling



Strategies

- Improving the understanding of parent-child relationships

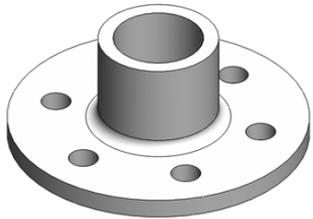


CATIA

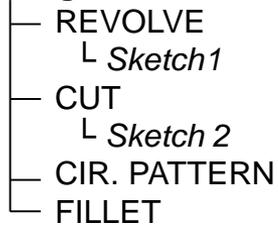


Strategies

- Improving the understanding of parent-child relationships



Flange



Design Tree

		Child			
		Revolve	Cut	Cir. Pattern	Fillet
Parent	Revolve		1	0	1
	Cut	1		1	0
	Cir. Pattern	0	1		0
	Fillet	1	0	0	

Dependency Matrix



Strategies

- Modeling methodologies

1. High Quality Modeling



Strategies

- Modeling methodologies



"I'M A STRONG BELIEVER IN CAD STANDARDS-THAT'S WHY I USE THIS EXCELLENT SET OF MY OWN"



"WE BELIEVE WE NOW HAVE A CONSENSUS ON THE CAD STANDARDS"

© Roger Penwill



Strategies

- Modeling methodologies

Horizontally-Structured CAD/CAM Modeling

Landers, D.M., Khurana, P., (2004). *Horizontally-Structured CAD/CAM Modeling for Virtual Concurrent Product and Process Design*. US Patent 6,775,581. Delphi Technologies

Explicit Reference Modeling

Bodein, Y., Rose, B., Caillaud, E., (2014). *Explicit Reference Modeling Methodology in Parametric CAD System*. *Computers in Industry* 65(1), pp. 136-147.

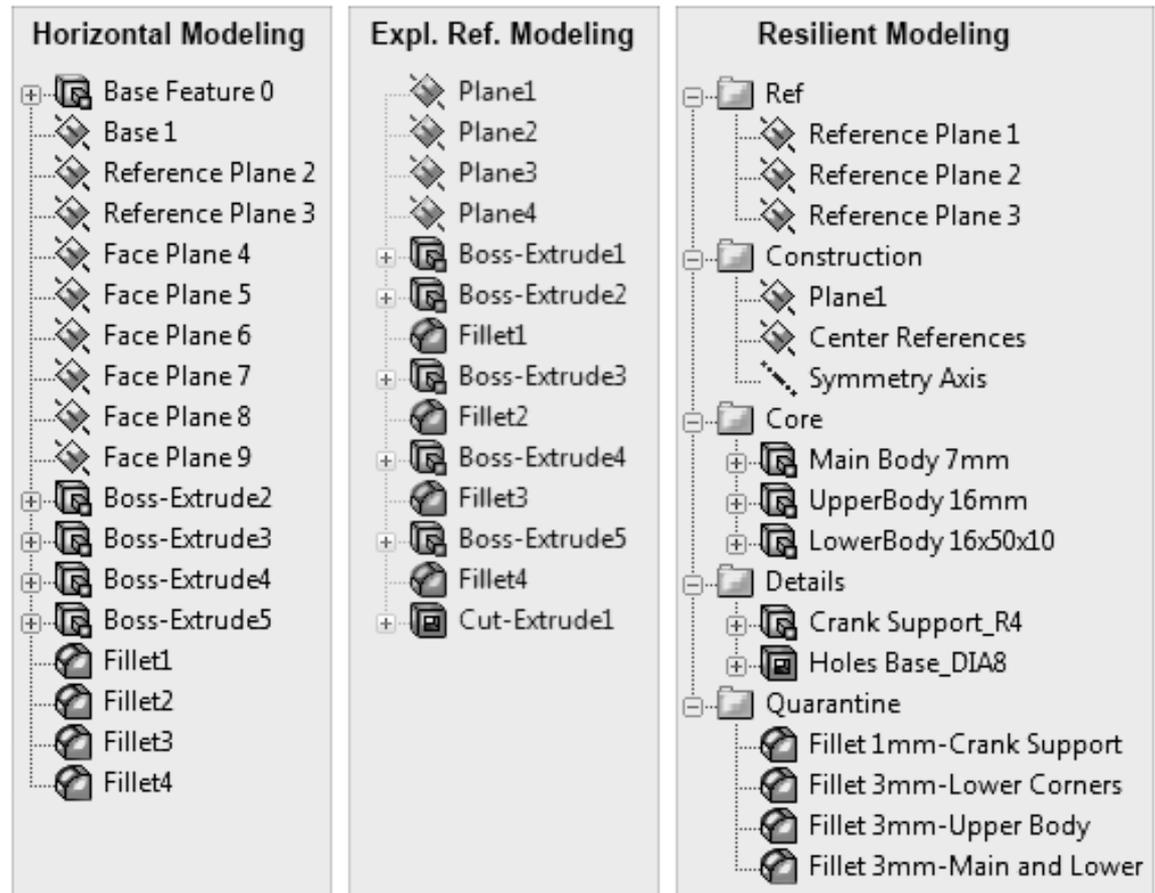
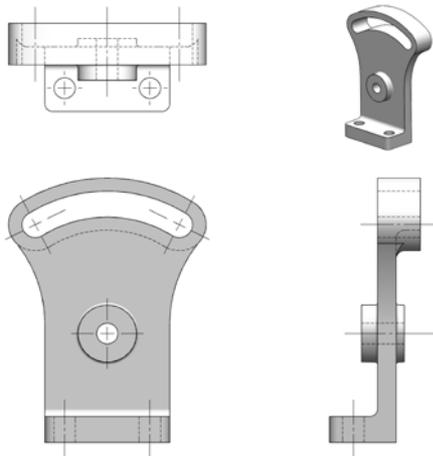
Resilient Modeling

Gebhard, R., (2013). *A Resilient Modeling Strategy*. Technical Presentation, Solid Edge University. <http://resilientmodeling.com>

1. High Quality Modeling



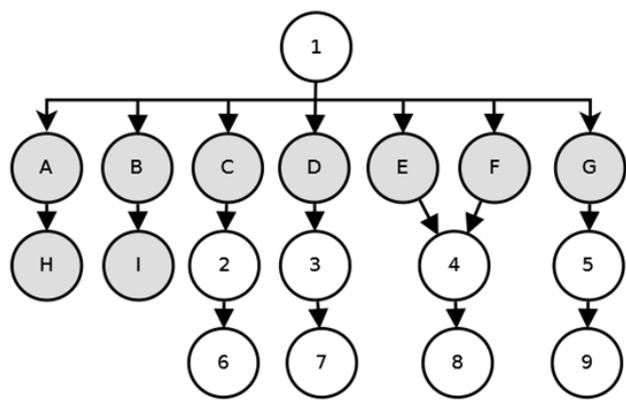
- Design trees by methodology



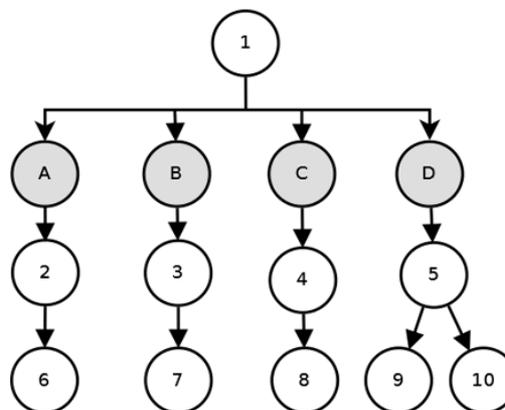
1. High Quality Modeling



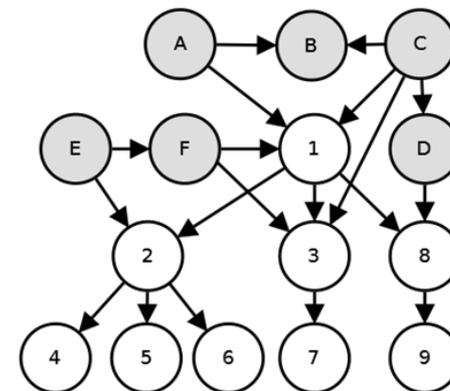
- Graph complexity and structure varies by methodology



Horizontal



Explicit References



Resilient

● reference geometry ○ model features



Contributions

- We have conducted numerous experiments to test and compare different modeling approaches in a variety of design scenarios and modeling situations .
- Levels of quality (Linguistic Model):
 - Morphological, syntactical, and semantic.
- Dimensions of quality:
 - Validity, completeness, consistency, conciseness, clarity, and design intent.
- The goal is to develop a defined set of quantitative metrics.

2. Model Quality Testing



© Roger Penwill

2. Model Quality Testing



Analyze, correct, and repair defects (particularly at the semantic level) is still necessary to guarantee the reliability and robustness of master digital models.

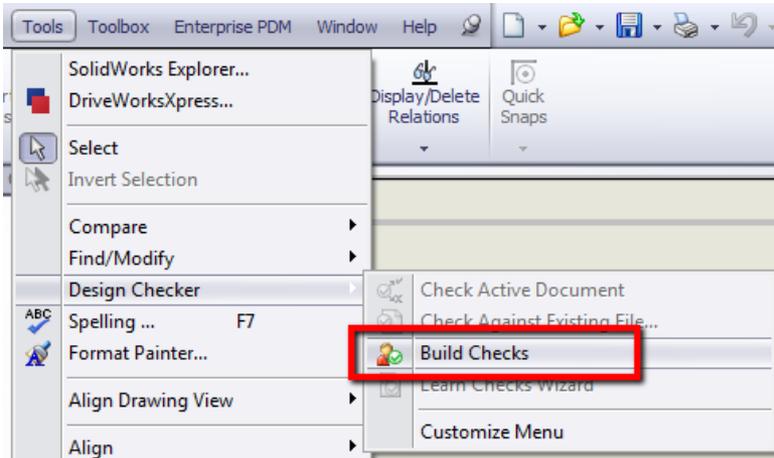
© Roger Penwill

2. Model Quality Testing

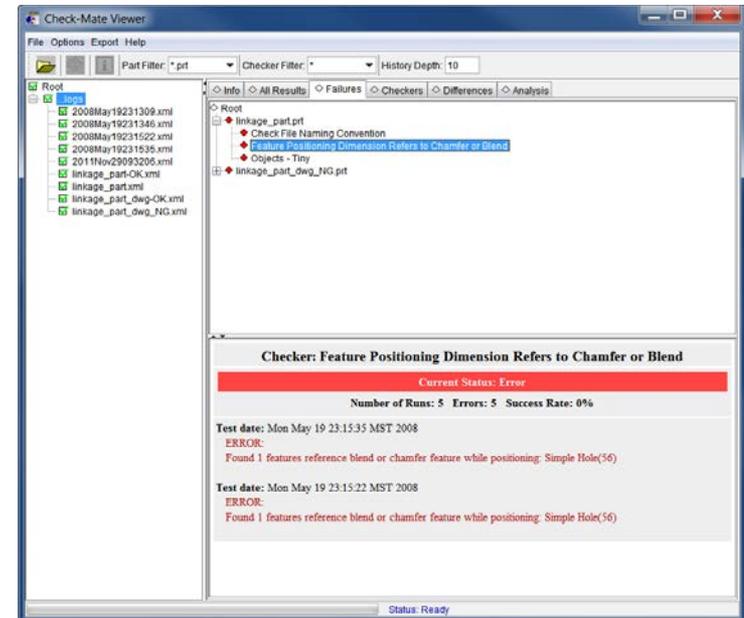


Model Quality Testing tools

- Quality testing tools for model verification, validation, and comparison are essential, as exporting CAD models that contain errors or anomalies to different downstream applications is prone to data corruption, which typically requires the models to be reworked by the downstream user.



SolidWorks Design Checker



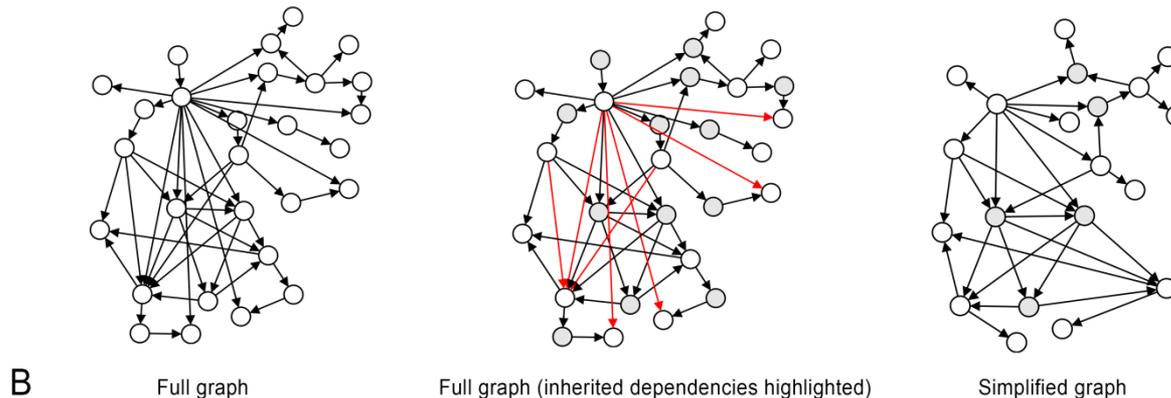
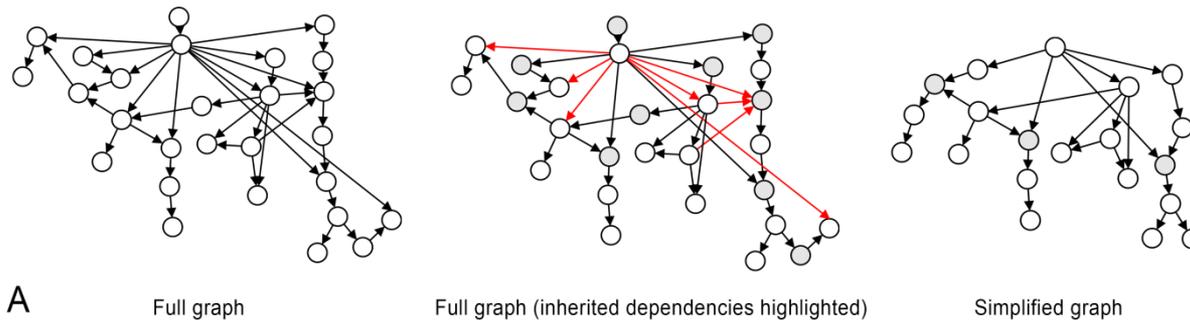
NX Checkmate

2. Model Quality Testing



Model Quality Testing Contributions

- Management of inherited dependencies in DAG.



2. Model Quality Testing



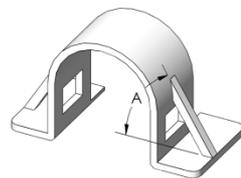
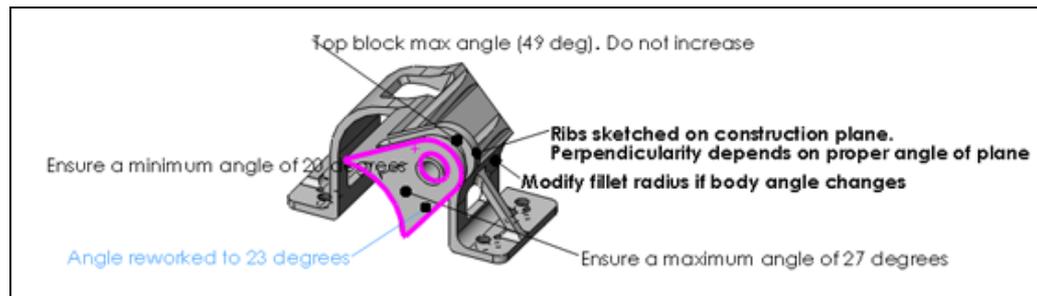
Model Quality Testing Contributions

- Detection of Dimensions of Quality.
- Automatic Repair and Correction.

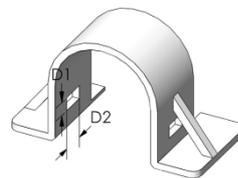
3. Model Enrichment



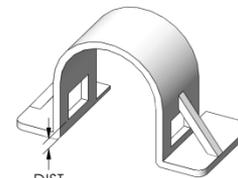
- Enrich models with structured design information and mechanisms to centralize knowledge and facilitate design communication.
- Annotated models provide significant benefit over non-annotated models in terms of reusability, especially when undergoing design alteration.
- Built-in historical record of design information.



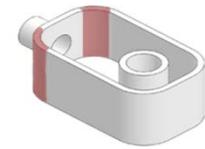
Solution 1: Increase rib angle



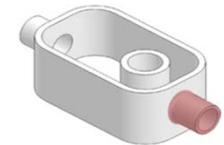
Solution 2: Change side cut dimensions



Solution 3: Move side cut down



ALTERATION 1
(fillet edges)



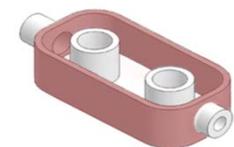
ALTERATION 2
(create outlet)



ALTERATION 3
(modify outlet)



ALTERATION 4
(create new connector)



ALTERATION 5
(modify width and height)

3. Model Enrichment



Challenges

- *Representation:* external vs. internal.
- *Content and Structure:* what needs to be provided and in what form.
- *Interface:* interaction system used to enter and retrieve annotation information.
- *Visualization:* visual clutter, information overload, etc
- *User Motivation:* incentives, documenting, etc.

3. Model Enrichment



Representation

- *Internal*: annotations stored within the CAD model
- *External*: annotations stored in external repositories

	Advantages	Disadvantages
Internal	<ul style="list-style-type: none">• Easy implementation• Full integration with the model (low maintenance)• Efficiency in terms of processing and manipulation• Already supported by most CAD systems	<ul style="list-style-type: none">• Original document changed• Difficulty for multiple sets of markups• Difficult to search annotations content
External	<ul style="list-style-type: none">• Multiple independent sets of markup• Progressively information update• Easy distribution of information in collaborative environments and over the web.• Information can be processed and analyzed separately	<ul style="list-style-type: none">• Difficulty of implementation• Problem of persistent references• File maintenance

3. Model Enrichment



Content and Structure

- What information should be included? How do we capture it?
- Structured vs. unstructured
- When is the user providing too much information?
- Can we develop a standard structure or language for annotations?

3. Model Enrichment



Interface

- Mechanisms to support user interaction with the information are needed.
- None of the current standards provide explicit guidelines about managing 3D annotations, although they do recommend the use of groups, layers, or views to make the model more readable.
- Interaction with annotations should be integrated with the CAD application (PMI modules) so designers do not need to leave the CAD environment to annotate the models.
- Effort (time) vs. Benefits

3. Model Enrichment



Interface in current systems

- Current systems: poor usability. Basic information grouping tools.
- No selective visualization.
- No filtering and/or search functionality.
- Disconnected from collaborative design tools.
- No traceability of information and how it evolves over time.
- No links with external documents and information systems.

3. Model Enrichment



User Motivation

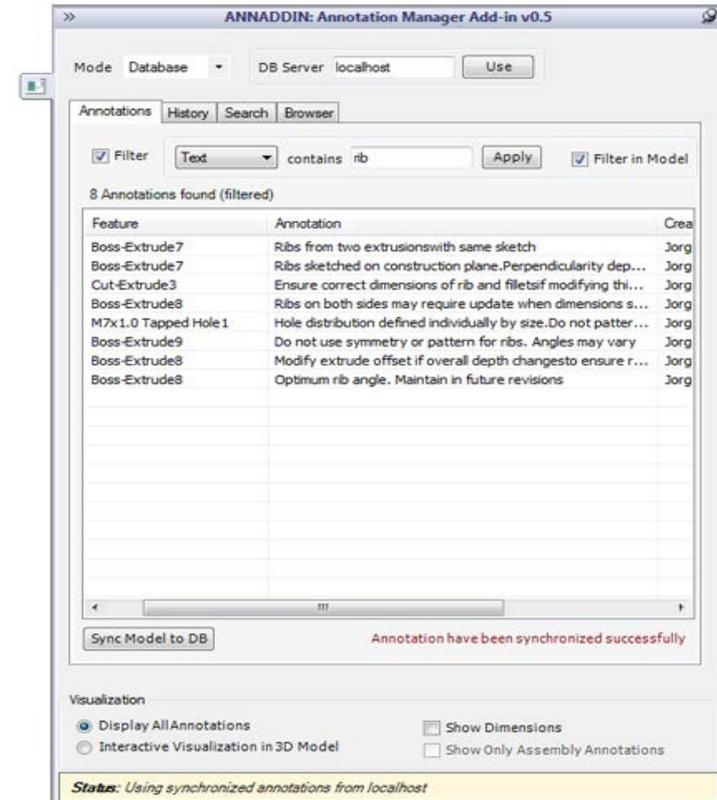
- Designers are reluctant to spend time annotating their designs
- The designer has no further use of annotations, as she already understands the design. Why should the designer do something that is only beneficial for people that come after her?

3. Model Enrichment



Contributions: Annotation Manager

- CAD Integration
- Enhanced User Interface
- Advanced visualization, filtering and search functionalities
- Interactive visualization via model features

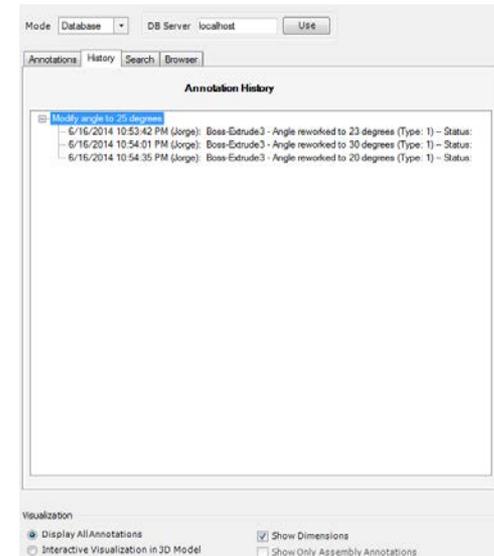
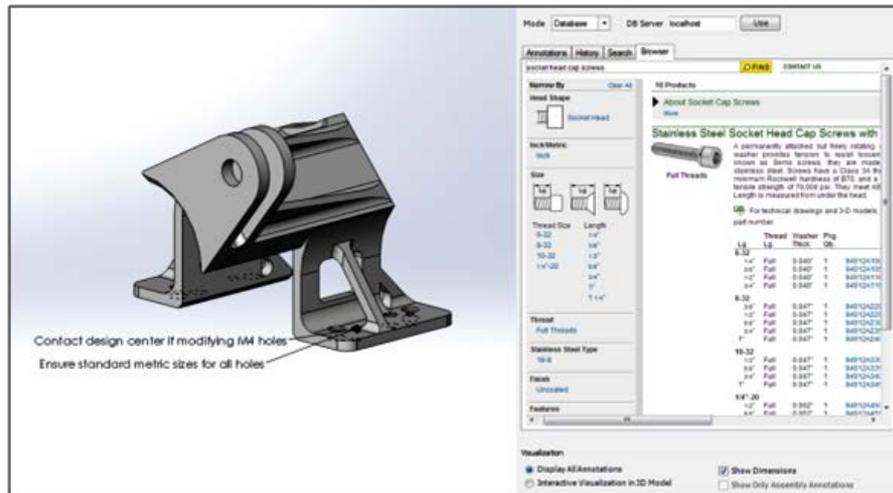
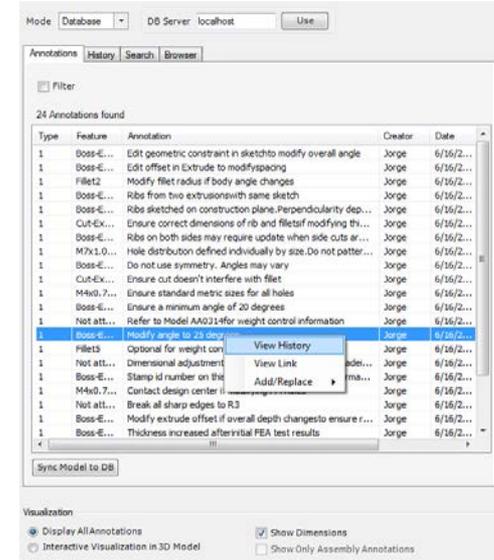


3. Model Enrichment



Contributions: Annotation Manager

- External document support (hyperlinks, multimedia content, etc.)
- PLM Integration
- Historical records and information evolution
- Videoconferencing



Thank You

Jorge D. Camba

jdorribo@uh.edu



Manuel Contero

mcontero@upv.es



Pedro Company

pcompany@uji.es



The icons used in this presentation are courtesy of José Sanmartín González, Delwar Hossain, Justin Blake, Hea Poh Lin, and Creative Stall from Noun Project (<https://thenounproject.com>)