Towards "Skinnable" Physical Objects

First Author
AuthorCo, Inc.
123 Author Ave.
Authortown, PA 54321 USA
author1@anotherco.com

Second Author
AuthorCo, Inc.
123 Author Ave.
Authortown, PA 54321 USA
author2@anotherco.com

Third Author
AuthorCo, Inc.
123 Author Ave.
Authortown, PA 54321 USA
author3@anotherco.com

Abstract
In this sample we describe the formatting requirements for various SIGCHI related submissions and offer recommendations on writing for the worldwide SIGCHI readership. Please review this document even if you have submitted to SIGCHI conferences before, some format details have changed relative to previous years.

Author Keywords
Optional section to be included in your final version.

ACM Classification Keywords
H.5.m [Information interfaces and presentation (e.g., HCI)]: Miscellaneous. See: `http://www.acm.org/about/class/1998/` for help using the ACM Classification system. Optional section to be included in your final version, but strongly encouraged.

Introduction
Motivation: we want to let non-technical people download pre-configured component-based-assembly circuits (e.g. LittleBits) and print out cases for them. For example, the Twitter bird. Or the weather statue.

Related Work
With the emergence of digital fabrication technologies, there have been a number of projects intended to simplify

the process of creating interactive objects.

Two recent projects considered building customized enclosures for exiting electronics. Weichel et al. created a system for designing laser-cut cases for projects created with the .NET Gadgeteer electronics prototyping platform [**?**].

Building enclosures: [**?**] (NatCut: An Interactive Tangible Editor for Physical Object Fabrication) [**?**] (Enclosed: a component-centric interface for designing prototype enclosures)

Stuff

Features:

- Represent the circuit as an undirected graph (currently input by hand in code).
- 2D packing to find the best arrangement.
- Minimize cost function of size, number of wires; something like $c = wS + (1 - w)W$ where S is size, W is number of wires, and w is relative weight between the two.
- Specify location of fixed modules (in code).
- Search all possible logical arrangments of wires/not wires.
- Output object for printing.