Prototyping with a Purpose

Brent M. Dingle
Game Design and Development Program
Mathematics, Statistics and Computer Science
University of Wisconsin - Stout

2014

Some content based on GDC 2006, Gingold and Hecker

Presentation Outline

- Prototyping General Info

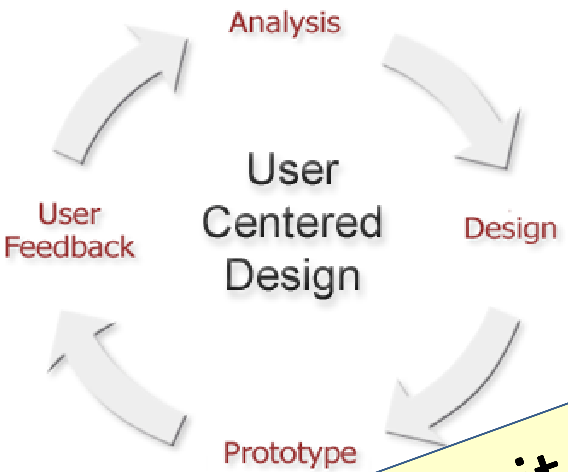
- Asking Good Questions

- How to Decompose a Project

- How to Measure a Prototype

Using Prototyping

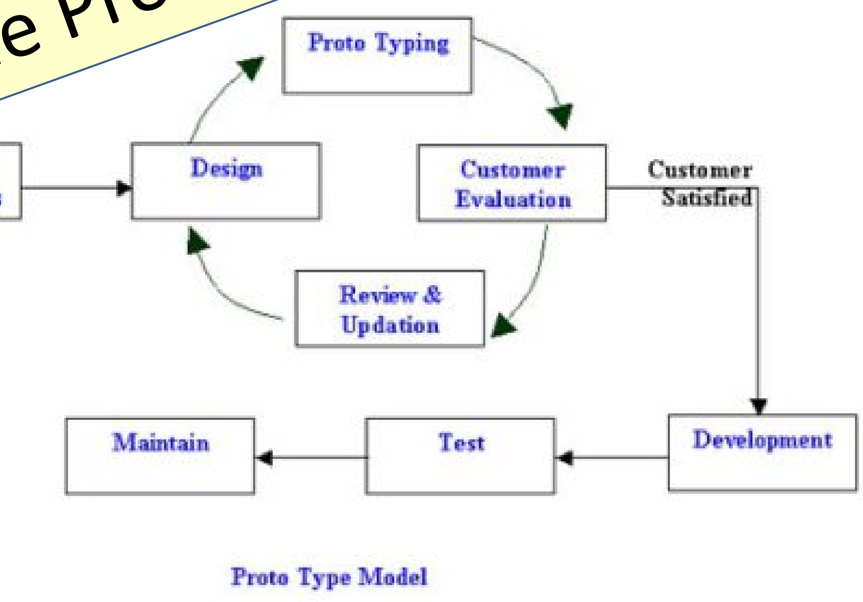
- Many ways to look at how Prototyping is used



Conceptually the same in every case

- Test something, Prove Something…
- Learn, Adapt…

Let's look at it from Game Production



Proto Type Model

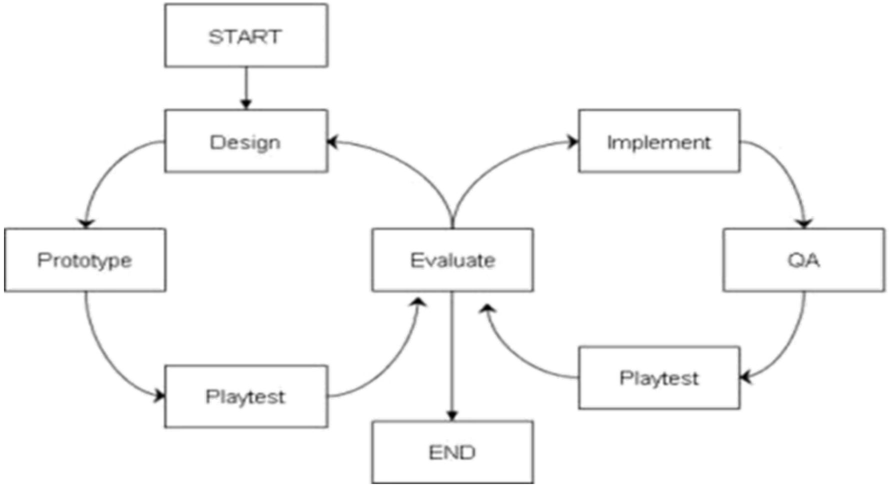
High Level Production Stages

- Have an Idea
- Discovery
- Preproduction
- Develop
- Sell

High Level Production Stages

- Have an Idea
- Discovery
- Preproduction
- Develop
- Sell

Prototyping
is useful and often necessary



Why Prototype?

- Most Common: Answer questions
  - e.g. Will this work?

YES this will work = PERSUASIVE

Prototype Validates the idea

- Also
  - Discover/uncover the un
    - downside an

What if answer is NOT yes, or not just "yes or no"

  - Persuade and inspire

Why Prototype?

- Most Common: Answer question
  - e.g. Will this work?

NO this will not work = DOWNSIDE

BUT it does do this and this = UPSID

- Also
  - Discover/uncover the unexpected
    - downside and upside

So what if we do it like this…

  - Persuade and inspire

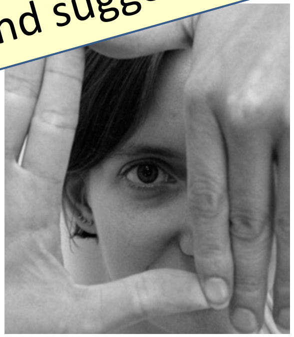
Why Prototype?

- Most Common: Answer questions
  - e.g. Will this work?

- Also
  - Discover/uncover the unexpected
    - upside and downside
  - Persuade and inspire

Prototypes do NOT GENERATE ideas
They VALIDATE them

But can generate upsides and suggest ideas



Presentation Outline

- Prototyping General Info
- **Asking Good Questions**
- How to Decompose a Project
- How to Measure a Prototype

Important: Ask a Good Question

- How to formulate good questions for prototyping

- Focus On
  - Where do you need understanding?
    - Target other questions in this area

Let's consider some examples questions…

Question

- Can we make a fun social game between characters?

Is this a good or bad question to use a prototype to answer?

Poor Question

- Can we make a fun social game between characters?

Bad!

There is nothing to try and test here.

What idea do you want to try out?

Question

- Is the
  “Leg User Interface concept”
  user friendly,
  powerful,
  and cool?

Is this a good or bad question to use a prototype to answer?

Good Question

- Is the Leg UI concept user friendly, powerful, and cool?

Good!

It is testable.

Build it.
Demo it. Ask people about it.
Do they need help? Is it cool?
Does it accomplish what we want it to?

Question

- Can rolling around a sticky ball be compelling?

Is this a good or bad question to use a prototype to answer?

Good Question

- Can rolling around a sticky ball be compelling?

Good!

It is testable.

Build it.
Demo it. Ask people about it.
Do they think it is cool and fun?
And do they keep playing?

Question

- Here is the game description document.
  - Is it going to be fun?

Is this a good or bad question to use a prototype to answer?

Poor Question

- Here is the game description document.
  - – Is it going to be fun?

Bad!

It is an idea.
But is very unfocussed.
It basically would require the entire game to be built,
which really is not a prototype.

You must deconstruct/decompose the features.
How do you reduce a large problem into smaller manageable ones?
And stay relevant to the big picture of the project's idea (it's vision)?

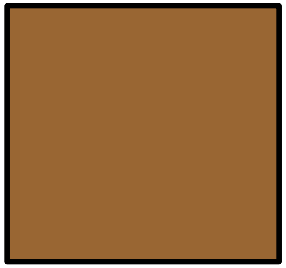
This is a skill to learn.

Presentation Outline

- Prototyping General Info
- Asking Good Questions
- **How to Decompose a Project**
- How to Measure a Prototype

Decomposition

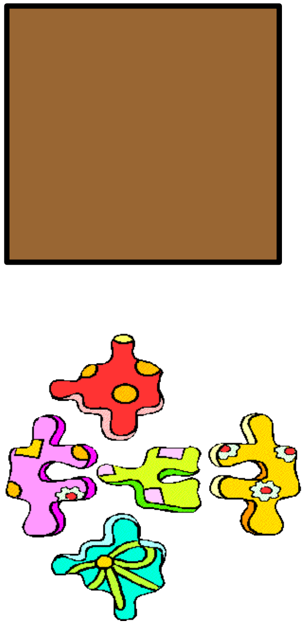
- Examine
  what you need to know about



Decomposition

- Examine
  what you need to know about

- Divide it into smaller pieces

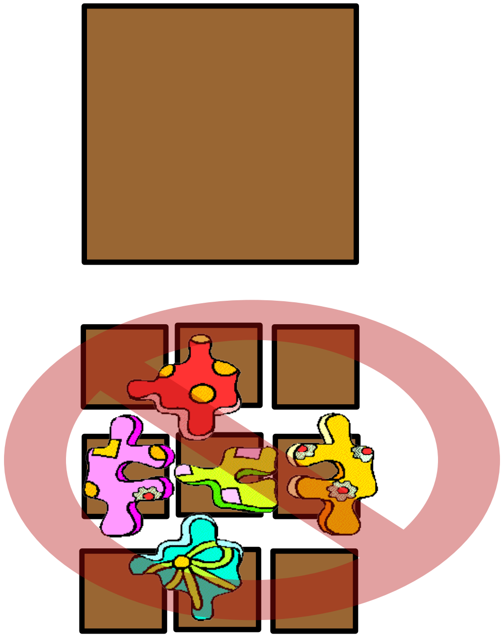


Decomposition

- Examine
  what you need to know about

- Divide it into smaller pieces

- Make sure the pieces fit together



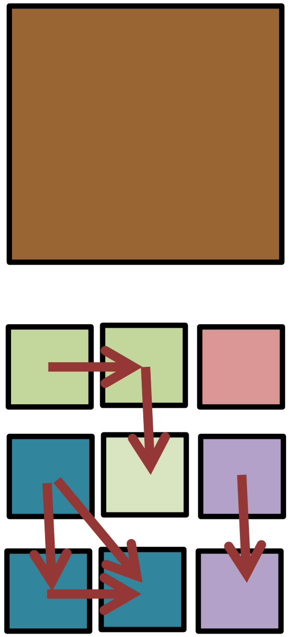
Decomposition

- Examine
  what you need to know about

- Divide it into smaller pieces

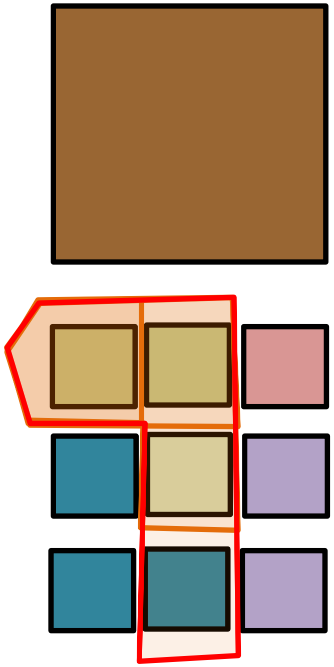
- Make sure the pieces fit together

- Keep track of which pieces
  depend on others



Decomposition

- Examine
  what you need to know about
- Divide it into smaller pieces
- Make sure the pieces fit together
- Keep track of which pieces depend on others
- Know the bounds of your prototype



Decent Size

Large Size

Huge Size

Out of Bounds

Decomposition

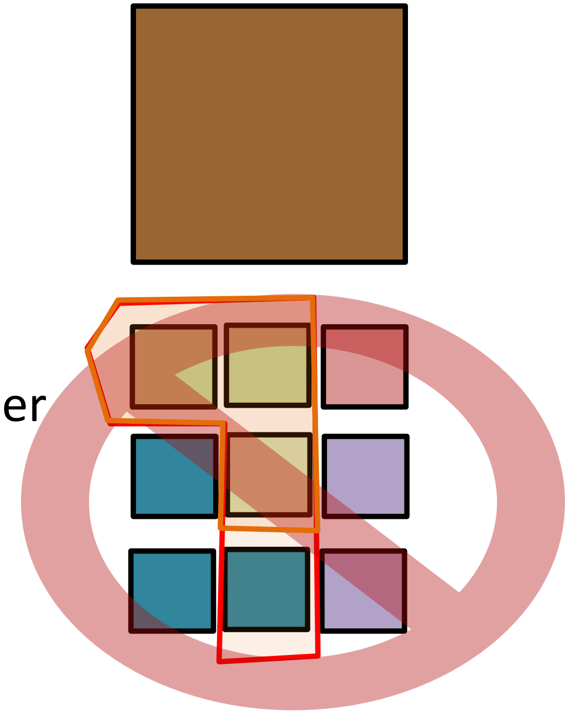
- Examine
  what you need to know about

- Divide it into smaller pieces

- Make sure the pieces fit together

- Keep track of which pieces depend on others

- Know the bounds of your prototype



Do not overreach
Stay inside the lines of your piece

Presentation Outline

- Prototyping General Info
- Asking Good Questions
- How to Decompose a Project
- **How to Measure a Prototype**

Measuring a Prototype

- Metrics
  - Cheap
  - Falsifiable
  - Relevant

Measuring a Prototype

- Metrics
  - **Cheap**
  - Falsifiable
  - Relevant

Take less effort than the real thing
Cost almost nothing (very little)

CHEAP & EASY

Measuring a Prototype

- Metrics
  - Cheap
    - **Agile**

Adding suggestions and changes should be trivial

  - Falsifiable
  - Relevant

Measuring a Prototype

- Metrics
  - Cheap
    - Agile
    - **Light**
  - Falsifiable
  - Relevant

Typically only one person working on a prototype at a time

We use teams of two for multiple educational reasons

Measuring a Prototype

- Metrics
  - Cheap
    - Agile
    - Light
  - **Falsifiable**

You should be looking to prove something.

You should be able to prove
a good idea is good
and a bad idea is bad

- – Relevant

Measuring a Prototype

- Metrics
  - Cheap
    - Agile
    - Light
  - Falsifiable
    - **Make a Claim**
  - Relevant

If you cannot explain why or what you want to achieve by making a prototype,

then you may want to rethink making it

The prototype should be needed to validate or disprove something

Example:

The [blarg] on the right side of the screen improves [shozbot] more than having it on the left side of the screen

Measuring a Prototype

- Metrics
  - – Cheap
    - Agile
    - Light
  - – Falsifiable
    - Make a Claim
    - **Testable**
  - – Relevant

How does your prototype TEST your claim?

Will it show something is clearly working or not?

Design the prototype so it obviously shows success or failure.
This is not a theoretical experiment.

And the results should not be “open”
to much variance in interpretation

Talk is cheap. Show me the code.
— Linus Torvalds

Measuring a Prototype

- Metrics
  - Cheap
    - Agile
    - Light
  - Falsifiable
    - Make a Claim
    - Testable
    - **Tested**
  - Relevant

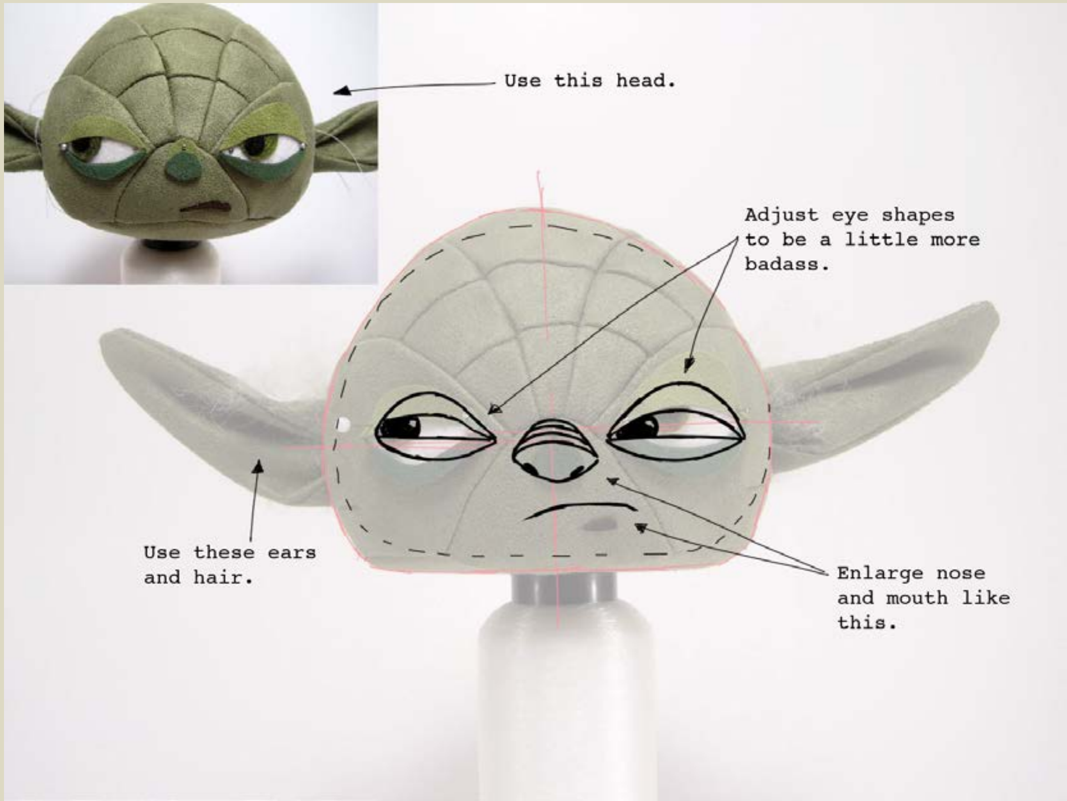
Tested by others
NOT just by prototype's creator(s)

Data collected

Explain why the design is this way
based on the data

Measuring a Prototype

- Metrics
  - Cheap
    - Agile
    - Light
  - Falsifiable
    - Make a Claim
    - Testable
    - Tested
    - **Learn**
  - Relevant


Use this head.
Adjust eye shapes to be a little more badass.
Use these ears and hair.
Enlarge nose and mouth like this.

Identify and Record:
What was learned from the prototype?

Measuring a Prototype

- Metrics
  - Cheap
    - Agile
    - Light
  - Falsifiable
    - Make a Claim
    - Testable
    - Tested
    - Learn
  - **Relevant**

Prototypes must be relevant to the project at hand

Generalizable helps

Prototype becomes incorporated into product
Code or Art can be reused
Design Reuse
Reference point for design/development

Measuring: In The **Bonus**

- **Surprising**
  - Feedback
  - Upside Downside
  - Inspiring

Good prototypes do something unexpected and useful

- Persuasive
  - Fun
  - Tangible
  - Clear
  - Disruptive

Measuring: In The Bonus

- Surprising
  - **Feedback**
  - Upside Downside
  - Inspiring
- Persuasive
  - Fun
  - Tangible
  - Clear
  - Disruptive

Good prototypes do something unexpected and useful

Good prototypes get lots of feedback

You want people to comment
and not just “yeah good” or “yeah bad”

If you have to beg to get meaningful feedback
Something is off
and it could be something major
that nobody wants to talk about
Like the whole concept stinks
or the prototype is “pointless”

Measuring: In The Bonus

- Surprising
  - Feedback
  - **Upside Downside**
  - Inspiring
- Persuasive
  - Fun
  - Tangible
  - Clear
  - Disruptive

Good prototypes do something unexpected and useful

Good Prototypes
should uncover GOOD and BAD things

If all goes as planned
with no exciting discoveries
or hidden pitfalls
the prototype is likely off in some way

Perhaps the wrong question(s) are being asked

Measuring: In The Bonus

- Surprising
  - Feedback
  - Upside Downside
  - **Inspiring**
- Persuasive
  - Fun
  - Tangible
  - Clear
  - Disruptive

Good prototypes do something unexpected and useful

A prototype should inspire MORE Questions
Revealing more fun stuff
or problems to investigate

WARNING: Eventually the continued exploring must end (be saved for later).
This hopefully happens when the problems to investigate are small and the fun is large.

If you are lucky you will always think it ends too soon.
If you are really lucky you will get to go back and do more (like sequel… oh yeah)

Measuring: In The Bonus

Good Prototypes should Convince people of things

- Surprising
  - Feedback
  - Upside Downside
  - Inspiring

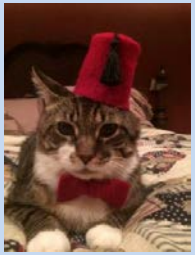
- **Persuasive**
  - Fun
  - Tangible
  - Clear
  - Disruptive

Measuring: In The Bonus

- Surprising
  - Feedback
  - Upside Downside
  - Inspiring

- Persuasive
  - **Fun**
  - Tangible
  - Clear
  - Disruptive

Good Prototypes should Convince people of things



The prototype needs to be
cool, exciting, entertaining, FUN!

That is what the game needs to be too
So in the background the prototype
tests more than just a single question

People must want to see and play
your prototype – just as they would the game

They should be excited about what you are doing

- Just like they would the game
- Again wanting to be involved, give feedback

Measuring: In The Bonus

- Surprising
  - Feedback
  - Upside Downside
  - Inspiring

- Persuasive
  - Fun
  - **Tangible**
  - Clear
  - Disruptive

Good Prototypes should Convince people of things

Prototypes are not theory.
They should CLEARLY communicate
and make REAL whatever they are getting at

People should look at the prototype and GET IT
The design, the concept, the fun, the interface, whatever.

Minimal explanation should be required.

Measuring: In The Bonus

Good Prototypes should Convince people of things

- Surprising
  - Feedback
  - Upside Downside
  - Inspiring
- Persuasive
  - Fun
  - Tangible
  - **Clear**
  - Disruptive

Prototypes are not theory.
They should CLEARLY communicate
and make REAL whatever they are getting at

People should look at the prototype and GET IT
The design, the concept, the fun, the interface, whatever.

Minimal explanation should be required.

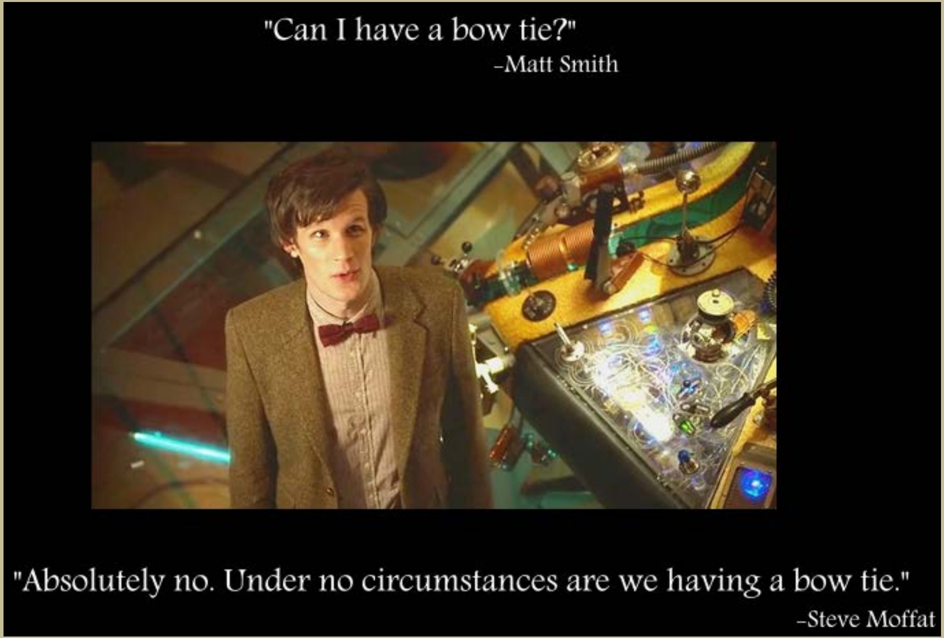
Hopefully that is clear

Measuring: In The Bonus

- Surprising
  - Feedback
  - Upside Downside
  - Inspiring

- Persuasive
  - Fun
  - Tangible
  - Clear
  - **Disruptive**

Good Prototypes should Convince people of things



"Can I have a bow tie?"
-Matt Smith

"Absolutely no. Under no circumstances are we having a bow tie."
-Steve Moffat

Prototypes should change people's minds

If nobody is convinced of anything…

Then what is the prototype really doing?

Keeping the Prototype

- Prototypes may be labeled “throw away”
- However
  - The results must be kept
  - to be able to Revisit and Reexamine the results
- A good prototype (and its results)
  - Will be consulted multiple times throughout development
    - If no one ever revisits it, then perhaps it was not such a useful thing to prototype
      - The answers it provided were never really in question?
      - Or the questions were not very useful?
      - Or it never really answered anything?

"Shall I file it or do you want to find it again?"

Theoretical Questions?

"Really? — the hokey-pokey ***is*** what it's all about?"

End

I'M BEING EXTREMELY CLEVER UP HERE AND THERE'S NO ONE TO STAND AROUND LOOKING IMPRESSED!

Additional Information/Resource

- GDC 2006: Advanced Prototyping Presentation by Chaim Gingold and Chris Hecker