**Mechanisms for Input**

To begin, I looked at other batch processing games. I looked at the approach they took to user input and split these into categories, compared them to each other and decided upon the pros and cons for each.

User defined input from a global list of ingredients:

BigPharma uses a system in which there is a number of input slots in the walls. The user can click on these input slots and choose a specific ingredient they wish to be output; this gives the user full control over what ingredients get output and *where* they are output from.

An advantage to this is that we give the player more freedom over their creation; control over where the input comes in gives the user more control over the layout of their production line and this ensures for more variety in solutions over a large player base. It also invites a further level of optimisation as it may be more efficient for certain ingredients to come out of certain inputs that are closer to each other and this therefore requires less space or means the solution requires less cycles.

Another advantage is that this introduces metagaming: in BigPharma, pills can be created using different ingredients and some ingredients may produce more favourable results than others (pills work faster, pills have less side effects, etc). By allowing the player to experiment with ingredients they not only have their production line to implement, but they can also optimise and improve their output by experimenting with different ingredients.

I believe a disadvantage is that this is theme dependant:

It only works with BigPharma because a lot of the game is about researching different ingredients and running experiments, like in a science lab, to create the perfect pill.

I don’t believe this would work as well with a theme idea such as our restaurant theme: meals can only be created using the required ingredients, and as such changing the inputs optimises only the physical layout of the production line.

User-determined input from a limited list of ingredients:

Whilst SpaceChem allows the users to determine their input from any ingredient available, we could limit which ingredients are available on each level, but still allow the user to decide where they want each ingredient to be input.

This keeps a level of optimisation and creativity that comes with choosing the best location for each ingredient to be input, but doesn’t overwhelm players with choice and thus this may be more suitable for casual players.

Pre-determined input on a per-level basis:

SpaceChem and Infinifactory use a system where the input is pre-determined and changes per level. On each level, the input that you need to complete the level is given to you but you do not get to decide from where; you must work with where it is placed.

This removes the experimentation from the game that users get with Big Pharma, whether this is a good or bad thing is to be decided. This may or may not have any impact based on the theme of our game.

Advantages:

An advantage is that this makes the game much easier for casual play / casual players. Our target audience consists of 20-40 year old females and males, with females making up almost half of the player base. Whether or not this audience would be interested in the level of optimisation and metagaming that comes with user-defined input is something we would have to look further into.

When playing Infinifactory, it’s very clear to see how the input must be transformed into the output and this makes it very easy for the user to understand their task.

Disadvantage:

This removes the level of variety of number of solutions for each level. When users decide what and where the input is, this adds for extra creative room and means that players are less likely to find the same solution. It also removes that extra level of optimisation that comes from optimising the location of your inputs.

Pre-determined input on a global scale (the same input for every level):

Pre-determined input on a global scale means that every level will have the same input. This is an idea for the elemental theme; everything can be created by starting with just the four basic elements of air, fire, water, earth.

This means we start with very small systems (earth + water = mud), and can eventually work up to very large systems comprised of many smaller sub systems.

Advantages:

Logical game progression. By creating your smaller sub systems, you can use these to create larger systems; the game progression can be seen by the player and the player can reuse their previous systems to now become a part of a larger system.

Introduces optimisation incentive. A player may use an old solution to be a part of a bigger solution, however realises this needs optimisation and so now there is an extra level of optimisation that is added to the game.

Disadvantages:

By having the same input on every level it can be argued that we are taking away a level of user creativity and taking away a bunch of variety from the list of possible solutions.

Summary / options:

**User-defined input from a global list of ingredients:**

*Pros:*

* Very high level of variety and user creativity
* More optimisation
* Introduce metagaming

*Cons:*

* May not work well with all themes?
* Is this suitable for casual gamers?

**User-defined input from a limited list of ingredients:**

*Pros:*

* Similar to the above option but more suitable for casual gamers
* Still have a level of optimisation

*Cons:*

**Pre-determined input placement (per level):**

*Pros:*

* Easier for casual gamers
* Clear to see objective

*Cons:*

* Less user creativity
* Less solution variety

**Pre-determined input placement (globally – same input for every level):**

*Pros:*

* System re-use (Larger system composed of many smaller sub systems)
* Optimisation becomes key

*Cons:*

* Less user creativity
* Less solution variety