Table of Contents

Design Concept

A design concept is a collection of embodiments that completely cover all the requirements of a design situation.

What is a design concept?

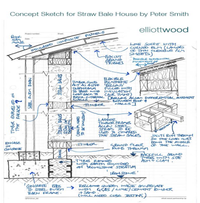
A *design concept* is a combination of embodiments that have been integrated into a coherent whole such that every system is contributing to the fulfillment of the requirements.

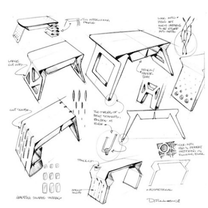
- Embodiments are like building blocks used to construct a variety of different overall concepts for interventions, only one of which will eventually be *the* team's concept.
- Embodiments cover only *some* of the requirements; a concept must cover *all* the requirements.

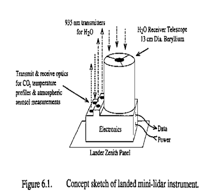
By definition, concepts are quite vague.

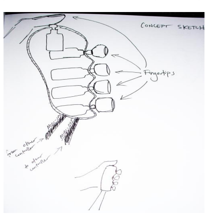
- You will not know the *precise* shape, mass, behaviour, colour, etc. of a design when it is still in the conceptual stage.
- But you will know the principles and technologies that are needed to eventually implement a suitable design intervention.

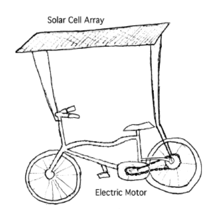
Design concepts are often described graphically with a sketch. Here are some sample design concept sketches.

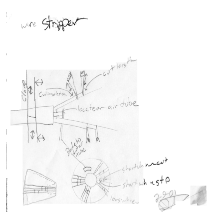












Notice how different they are from one another.

- Some are much more polished than others, but they're all equally valid as design concepts.
- The sketches themselves are relatively meaningless on their own.
- They become significant when all the other information about the design (the PRS, PAS, etc.) is available; the design concept stitches all this information together setting a direction for a final design intervention.

Just because many design concepts are rendered as having shape doesn't mean the final intervention's shape must be the same. Even at this stage, the structure of the design must submit to the needs of behaviour and function, and will change to ensure behaviours and functions are available in the best possible ways.

Design of complex interventions is a process of gradually seeking out the best of very many possible design interventions.

- It is easier, faster, and cheaper to develop and evaluate design concepts than it is to develop and evaluate many different fully detailed designs.
- Even though concepts are vague, it is possible to distinguish between those that are likely to be suitable and those that are likely to be unsuitable.
- The number of unsuitable designs is almost always much, much larger than the number of suitable designs, so even a qualitative assessment - based on concepts rather than fully detailed designs - is more effective and efficient at zeroing in on the “best” designs in the long run.

How do we develop design concepts?

A morphological chart implies a large number of overall concepts. We call the set of all these concepts a *design space*.

Our job at this point in the design roadmap is to try to find the best concept in the whole space. This is usually an intractable task.

There are many ways to manage the complexity of exploring a design space. To keep things simple in this course, we will follow only one process, described below.

Remember: As you work on your concepts, you may find there are unanswered questions about the design brief or your goals that require doing further research.

If this happens, it's vital that you document that research in your situation scan.

Step 1: Select preliminary concepts

The goal of this step is to identify a few concepts from the morphological chart (MC) that a team believes are likely to be reasonable places to begin developing a full intervention.

Each team member will:

- Review the Persona and SUC for which they are responsible.
- Select from the MC embodiments that are “best” for their Persona situated in the SUC.
- Combine those embodiments into a single overall concept.
- Document the preliminary concept and the justification for the selection of embodiments.

Step 2: Refine preliminary concepts

Refining you concept involves two tasks: *identifying interaction errors*, and *updating the concept* to correct the interaction errors.

This step must be performed at least twice. That is, each team member with identify some interaction errors, then update their concept, then search for more interaction errors, then update their concept again.

Identify interaction errors

Given a concept per team member (from Step 1, above), then next step is to refine that concept by thinking about how the product will be used by a given Persona in a given SUC, and looking for interaction errors (IEs) between the concept and the users.

This step is done individually, each team member working on their own concept.

Think of watching a movie in which your Persona is using your concept in your SUC. Don't forget to consider the setup and put-away stages that precede and follow the actual use of the concept.

- Watch every single thing your Persona is doing.
- Think also of any co-users or other people in the SUC who may distract or otherwise influence your Persona.
- Consider each interaction the Persona has with the concept.
- What kind of mismatch might occur in that interaction?

The mismatches you identify are IEs. Document each one. Identify at least two or three distinct IEs *throughout* the “movie” of your concept's usage. Read more about interaction errors.

- You should be able to find *many* IEs for a preliminary product.
- We do **not** expect you to find **all** of them.
- We **do** expect you to find significant ones throughout the entire usage.
- By showing you can find varied, representative, and significant IEs, you are demonstrating sufficient mastery for scholastic purposes. Finding IEs that are not sufficiently varied, not representative of all steps of usage, or not significant is indicative of insufficient mastery.

Update the concept

Now, given the interaction errors, modify your concept to address those errors. Remember, an IE is *always* an error on the product's side of an HMIL.

IEs will arise from mismatches between what one or more of your embodiments can do, and what users expect.

- You cannot change humans, so you cannot change your target users or expect them to behave differently than they otherwise would.

- Instead, you must change how embodiments work to interact better with the users.

This can be done in two ways.

Option A: Adjust an embodiment

You can change position, orientation, general size, general shape, weight, texture, or any other attribute of the embodiment.

Example 1: Designing a way to dispense food and drink, including coffee, on aircraft.

- Your original concept places a coffee maker, including a rather large tank of hot water, at the top of a cart. You did this to prevent the flight attendants from having to bend repeatedly to reach a coffee maker positioned low on the cart.
- You subsequently identify an IE: the most ergonomic location of handles to push the cart is such that the resulting forces may cause the cart to tip over because of the heavy water tank at the top of the cart.
- You revise the *coffee system* so that the hot water tank is located at the bottom of the cart while the coffee dispensing subsystem remains at the top of the cart. This changes the weight distribution and improves stability without adversely affecting the ability of flight attendants to reach for coffee easily.

What else might you have done to improve the concept?

Example 2: Designing a food blending system.

- Your original concept includes buttons to select the speed/mode of operation of the device. The buttons are positioned such that an elderly person can easily press the correct button.
- You subsequently identify an IE: an elderly female user is preparing smoothies for their grandchildren (co-users), but the grandchildren are nagging the grandmother, distracting her. You realize you had not considered a *distracted* user when you arranged the buttons.
- You revise the size and separation of the buttons to decrease the chances of a distracted user pushing the wrong button.

What else might you have done to improve the concept?

Option B: Replace an embodiment

You may not find any way to alter an embodiment to correct for an IE. In that case, go back to your team's morphological chart and look through the alternative embodiments for the system in question. Select another embodiment and reconstruct your concept to include the new embodiment that you believe will address the IE in question.

Important notes

You may conceive of a new embodiment that had eluded you during the ideation step. If this happens:

- Review the new embodiment with your team.
- If everyone agrees that it's a good embodiment, add it to the morphological chart and document the addition, paying particular note of the circumstances by which you conceived of it.
  - Documenting “acts of creativity” in real life is important for matters of intellectual property.
  - Adding the new embodiment to the MC makes it available to your teammates as well.

Step 3: Integrate preliminary concepts

For this step, all students are expected to study a 22-minute video of a "Deep Dive" design by IDEO. In particular, pay attention the part of the video about how the individual concepts were combined.

Once all team members have performed two iterations of refinement on their individual concepts, the team will work together to create a single concept embodying the best features of all the individual ones.

The goal is to create a single concept that can satisfy the needs of **all** the Personas in **all** the SUCs.

If you've executed all the steps properly, then *all* the concepts will satisfy the requirements of your project. In that case, combining the concepts should be relatively easy.

- It is generally easier to combine two concepts for, say, small commuter aircraft than it is to combine a concept for a small commuter aircraft with a concept for a 500-person trans-Pacific airliner.

Here's how to execute this step:

1. Hold a meeting with your team.
2. Share all the individual concepts with all team members.
   - Be sure to include which Persona and which SUC are attached to the concept.
3. For each individual concept in turn, collaboratively decide what the key features of the concept are that are most relevant to the Persona in the SUC.
   - The team member who developed the concept under discussion should be considered the “expert” on that concept, and should be expected to respond cogently to questions about why particular embodiments, shapes, sizes, materials, textures, etc. were chosen.
4. Collect the key features into a single list. These are the features that need to appear in your final concept.

    - Check the list of features for duplicates or for features that conflict with one another for some reason (e.g., power requirements, size or capacity, etc.). The team needs to remove duplicates and resolve conflicts.
5. Collaboratively formulate a single concept that embodies all the features in the list. This is your final concept.
    - This final concept must (a) satisfy all the requirements, and (b) satisfy the needs of all the Personas in their respective SUCs.

IMPORTANT NOTES

- You may find you are unable to integrate the individual concepts due to some incompatibility that only became evident once individual concepts were compared.
    - In this case, you will have to figure out what embodiments are responsible for the incompatibility and use the techniques in Step 2, above, to fix the incompatibility.
- You may end up with a final concept that looks *nothing* like any of the individual concepts. This is not a problem.
    - Example: modern smartphones emerged from the combination of the PDA and the mobile phone, yet smartphones really don't look like either.

Step 4: Build usage scenarios of final concept

Given a final design concept, each team member will create a usage scenario describing how their Persona would use the concept in their SUC. See the page on usage scenarios for details.

There will thus be as many usage scenarios as there are members in a team.

Step 5: Refine final concept

This step is similar to Step 2, except that now you all work collaboratively in your teams to refine the final concept.

Specifically:

1. Working together, look for several interaction errors in your final concept by analyzing the usage scenarios.
2. Working together, look for ways to address the interaction errors of the final concept to improve the concept.

Remember to document the interaction errors you identify in this step just as you did in Step 2.

OLD STUFF BELOW TO BE HIDDEN

HERE ⇒

In this step of our design process, we want to identify at least some of the most likely suitable concepts that are implied by the morphological chart. (Or, conversely, we want to eliminate the least likely concepts, some of which have already been identified when we searched for inconsistent embodiments.)

Assuming you have executed ideation properly and generated an appropriate morphological chart, then **a design concept is just one embodiment chosen from each row of the morphological chart**. Each row of the morphological chart gives embodiments for a given system. If you have implemented every system, then you have covered every requirement. Thus, a design concept selected in this way will cover all the requirements for your design.

That is, assuming ideation is used, generating design concepts is *not* a creative task, but rather an analytic search task.

In the end, out of the hundreds or thousands of concepts implied by a morphological chart, we want no fewer than five, and preferably around 10, design concepts, the suitability about which we are quite confident. These 5-10 concepts will be evaluated more closely in concept evaluation stage - but for now we can use a far more qualitative approach. We can use a qualitative (i.e., “quick and dirty”) approach because the differences between the suitable and unsuitable designs will be so stark and obvious that we don't need a detailed and labour-intensive method to separate them.

There are two such broad methods for searching a morphological chart for suitable design concepts: the “brute force” method, and the “hill climbing” method. Each is described below.

A third, semi-automated method is also described, which provides Google-Sheet-supported automation for some steps of the brute force method.

Brute force search

The brute force method exhaustively checks *every* concept represented in a morphological chart. While this method guarantees that you will consider every concept in the morphological chart, it is only feasible if the morphological chart is relatively small. If it takes about two minutes to identify and qualitatively evaluate a given concept[1)] from a morphological chart, then applying this method to a morphological chart that captures 200 concepts will take just under 8 hours (roughly one “business day”).

NOTE FOR STUDENTS. If your morphological chart has more than 200 concepts, you should probably use the Hill Climbing method described in the next section.

Here is how you would execute the brute force search.

When we “identify” a concept, we mean that you write down the embodiments, by their ID (see the example below) that constitute it. Furthermore, spend a little time considering how the embodiments would fit together into a single design intervention. This is best done by producing a very rough sketch that shows how the embodiments would likely fit together.

Also, we can create a linear list of all concepts by beginning with the concept that includes the “first” embodiment for each system, then iteratively changing each of the embodiments in turn.

When we “compare” two concepts, we are considering them as potential designs to satisfy the requirements. That is, you must answer this question: given the requirements, which of the two concepts being compared is more likely to do so well? Remember that you must be able to justify your selection of one concept over the other; clearly everyone on the team must agree on these justifications.

It is essential that you have the PSS, the PRS, the PAS, and your inconsistent embodiments available during this process.

1. Identify the first 10 concepts. Write them down.
    - This list will keep changing as you replace embodiments. At the end of the process, the list will contain the 10 best concepts, based on your qualitative analysis.
    - You can do this in any order; it doesn't have to be random.
2. Identify the next concept.
3. If this concept includes inconsistent embodiments, discard it and go to Step 2.
4. Compare this concept to each of the concepts already in the list.
    - If you find a concept in the list that is worse than the new concept, replace the concept in the list with the new concept.
    - If there are multiple concepts in the list that are worse than the new concept, then replace the worst of the worst concepts with the new one.
    - If you get to the end of the list and find no concept in the list that is worse than the new concept, discard the new concept.
    - Document which concept was discarded, noting the shortcoming in the discarded concept as the reason for discarding it.
5. Go to Step 2, and continue this loop until all concepts in the morphological chart have been examined.

Hill-climbing search

If your morphological chart represents tens of thousands of millions of concepts, it simply isn't tractable to use the brute force method. Even if you have an extensive list of inconsistent embodiments, you'll still have thousands of concepts to evaluate.

Since you cannot review the entire “space” of design concepts the best you can do is to *sample* it. That is, you *could* generate a random selection of all of the concepts[2)] - say 200 of them - and then apply the brute force method to those 200.

There's a problem with this approach, however. The random sampling process may well select unsuitable concepts that are quite close (in the “space” of concepts) to a very good one - and you'd never know it. Random sampling of a design space is, therefore, not a good way to find suitable design concepts.

Another approach could be to pick one concept at random, then iteratively improve it by comparing it to other concepts that are not very different from it (i.e., comparing two concepts that are different with respect to only one or two embodiments and choosing the better of the two). If you keep doing this, you should eventually reach the best concept in the whole space without having to check every concept. This is the essence of so-called hill-climbing searches.

The problem with this approach, however, is that there may be “local maxima” - concepts that are quite good but are separated from even better concepts by areas of bad concepts. Since hill-climbing only seeks better solutions, it will get stuck at the local maximum and not be able to proceed to the other, better solutions. (See the Wikipedia entry on hill-climbing for more information about local versus global maxima.)[3)]

We need to trade-off the amount of work you have to do to search the design “space” (which we want to minimize) against the likelihood of finding the “best” concepts (which we want to maximize).

A good compromise approach in this case is to select 10-15 concepts at random, and then apply a hill-climbing search to each of them. While there is no guarantee that this approach will find the best concepts in the space, it *is* likely that they will all be above average. And this approach will also be very time-efficient, compared to other methods.

Here's how you run this method:

1. Generate 10-15 *random* concepts from the Morphological Chart. For **each** of these concepts:
2. Make sure the concept is consistent; that is, make sure it contains no inconsistent embodiments.
    - If the concept is inconsistent, replace it with another randomly selected concept.
    - Document these 10-15 random concepts by identifying which embodiments make them up. Include a rough sketch of what the concept might actually look like.
3. Consider the suitability of the concept with respect to the requirements and system architecture. Identify one problem with the concept. The problem may be functional, physical, ergonomic, etc.
    - Note that problems will normally manifest as problems *between* two embodiments, such as how two embodiments interact, but problems can also exist within single embodiments (e.g., an embodiment will be too heavy to work in a particular design).
    - Remember to refer back to your PRS and PAS. Think of how the concept may satisfy (or not) the requirements. Problems you identify *must* track back to a requirement or system interface.
    - If you find a problem that you *cannot* track back to a requirement or system interface, then you may have made a mistake in your requirements or systems design. You'll have to go back, find, and correct that error before proceeding.
    - Document the problem briefly.
4. Using the morphological chart, replace one or both of the problematic embodiments such that you think the modified concept will not have the same problem.
    - Update your sketch of the concept (or produce a new one) that shows the new embodiment in the overall concept.
    - Document why the revised concept is “better” than the problematic one.
5. If you were unable to find a problem with the concept in Step 3, save this design concept and go on to the next of the 10-15 random concepts.
6. If you *were* able to find a problem with the concept in Step 3, go to Step 2.

You should execute Step 2 at least 5 and preferably 10 or more times for each of the initial randomly identified concepts.

Coming up with the first batch of 10-15 concepts is best done in your teams, so that everyone agrees that the batch is a representative random sample of the entire concept space.

The rest of the hill-climbing process can be done individually. That is, each team member can take 2 or 3 concepts and perform the hill-climbing process on their own. This can parallelize (and thus speed up) the overall process.

The result of this process will be 10-15 concepts, each of which has been refined by successively looking for and addressing problems in them. The documentation for each of these concepts should include:

- the original sketch of the concept and a list of the embodiments used;
- a list of
    - each of the problems found, in order,
    - each new embodiment swapped in to the concept to improve it, and
    - a rationale for the swap;
- a final sketch of the concept and a list of the embodiments used.

Semi-automated search

This method depends on a tool written as a Google Sheet to help manage the search. It can be used to implement a brute force search, or to help seed the hill-climbing search.

A complete tutorial on the tool is available at mec325_rdcg_tutorial.pdf.

The Google Sheet itself is available here[4]. **You must make a copy of this Sheet before trying to use it.** If you "break" the spreadsheet, just download a fresh copy.

SPECIAL NOTE 1: When you load the spreadsheet for use, look for the `MEC325` menu in the menu-bar. If you do not see it, try reloading the spreadsheet.

SPECIAL NOTE 2: The "brute force" method will not run if there are more than 200 concepts total - this is just to keep the sheet from taking up too many resources in your browser.

Deliverables

The deliverables from this part of the process are as follows.

1. A brief statement of whether you used the brute force or the hill-climbing search, and why.
2. Regardless of whether you used brute force or hill-climbing, include the documentation described in the pertinent section above.
3. Whether you used brute force or hill-climbing, a list of the final 5-15 concepts left at the end. Each concept should be represented by a sketch of the concept roughly showing all the embodiments, and a brief verbal description (100-200 words) of the concept.

However

It is vitally important to be able to justify every decision you make in this stage of the process.

If you find you have a difficult time evaluating concepts as described above, seek the assistance of your instructor. You are expected to perform these evaluations reasonably well. It is up to you to make the time to approach your instructor for help.

This step of the process is best done in one, long sitting. It may be difficult to organize the time, but you will likely work faster and better if you can do the entire exercise in one (preferably) or two sessions. Do not work on this in short bursts punctuated by distractions or other tasks - it is likely to lead to terrible results.

Examples

Consider the example discussed in the ideation page and reproduced below. We will consider the brute force search in this case. This is not a full example, but it is intended to show some aspects of the method.

Example problem Design problem: You have to design a joint for a detachable robotic arm. It's basic systems are:

1. Structural system (to resist physical forces during operation)
2. Connection system (to attach and disconnect automatically)
3. Power system (to transmit power)
4. Data system (to transmit data)

Here are some ways to embody these systems. (This can be thought of as a textual version of a morphological chart. However, do not think that *you* can just list embodiments this way. *You* have to create a proper morphological chart.)

A. Structural system

1. Bolt and lug
2. Concentric rods/shafts
3. Cup and socket (like, say, your shoulder joint)
4. Meshed teeth in compression (as in the Canadarm; see image.)
5. Motorized hooks
6. Magnetics

B. Connection/disconnection system

1. Mechanical gearing
2. Solenoid/electrical actuators
3. Hydraulics

C. Power system

1. Space-rated electrical wire with mechanical connector
2. Space-rated electrical wire with magnetic connector
3. Induction transmission

D. Data system

1. Space-rated electrical wire
2. RF transmission (as in radio)
3. IR transmission (as is a TV remote)
4. Microwave transmission
5. Laser (optical fibre) transmission

This set of embodiments represents 270 total concepts. So we could use the Brute Force method in this case.

We start by taking the first 10 concepts that do not have inconsistent embodiments. Normally, this would include concepts A1-B1-C1-D1 through A1-B1-C2-D5.

However, say we have decided that C2-D2 would be an inconsistent embodiment because of the RF noise between the magnetic connector in the power system and the RF receiver in the data system. So we remove concept A1-B1-C2-D2 and replace it with the next concept, A1-B1-C3-D1.

Now we consider the next concept, A1-B1-C3-D2, but we have C3-D2 also listed as an inconsistent embodiment, so we discard it and go on to A1-B1-C3-D3.

We then compare A1-B1-C3-D3 to each of the 10 concepts currently in our list and look for a concept that we believe will be less suitable than it, with respect to the requirements.

Since all the concepts so far have A1 and B1 in common, we can start by focusing on C3 and D3. While induction transmission of power and IR data communications both have disadvantages in space application[5)], they pose a distinct advantage together. In conjunction, they remove all need for direct, fixed connections between the robot elements. This will significantly reduce the physical complexity of the connector that we are designing because with C3-D3. Reduced complexity (i.e., increased simplicity) will generally reduce cost and increase reliability. So concept A1-B1-C3-D3 has certain merits.

In this example, we cannot really judge which concept in our list is the worst compared to A1-B1-C3-D3 because we do not have a precise set of requirements. (This underscores how important it is to have a crisp and complete set of requirements during this stage of the design process.) Still, we may argue that concept A1-B1-C1-D1 is significantly worse than A1-B1-C3-D3 because the former will require higher precision connection (especially during the connection and disconnection operations) than the latter.

So we (a) discard A1-B1-C1-D1 for A1-B1-C3-D3, and (b) note why the latter is preferred to the former.

NOTE TO STUDENTS. You *will* have a usable set of requirements, so you *will* be expected to judge much more precisely why a given concept is better or worse than some other concept.

We continue in this fashion until we have checked every possible concept.

creativity, method, process, concept

1)
It is possible to identify and evaluate concepts at this rate, but it requires a lot of teamwork and attention. You will likely be exhausted after a whole day of this.

2)
Indeed, this can be automated with less than 20 lines of Perl or Python or any other modern

programming language.

3)

While there are algorithms for hill-climbing that overcome this problem, those algorithms require being able to automatically determine a single value of “suitability” for each item being searched. We cannot do this in a computer for design concepts - the evaluation is based on human skill, knowledge, and subjective information.

4)

Original slightly buggy version is here

5)

Induction transmission has significant losses compared to direct connections, and IR transmission in space can be affected by the severe temperature differences between sunlit and shaded regions.