SE464 Lecture (May 18, 2011)

**About Lab One**

**Question: Which is better – visitor or interpreter?**

**Answer:** Neither or both – it depends on the context.

**Question: What did you (the professor) consider by “cheating” design?**

**Answer:** An example is using Jython. If this was a programming course, you would not be able to hand the Jython solution. But for this course, it’s fine. However, it’s much harder to modify and extend the Jython solution for different requirements/specifications.

**Design Space**

The term "design space" refers to a set of possible design alternatives. A design space may be characterized formally or informally. Some ways of characterizing a design space formally include the Pareto Front (which requires formalizing the metric or objective space) or a morphological analysis (which formalizes the design decisions).

What is the relationship between designs in a design space? Are they totally ordered, like the integers? Usually not. It is more common that the designs in a design space are partially ordered, and that there is no distinguished supreme design in that ordering.

In a total order, any two elements can be compared and the result of that comparison is either <, >, or =. In a partial order a comparison may result in <, >, =, or ?: i.e., there may be pairs of unequal elements for which we cannot determine which is better than the other.

For example, in Lab 1 we looked at four designs for a simple calculator program. None of these designs is strictly better than the others. They all represent different trade-offs.

**Partial Order / Hasse Diagrams**

**Question: What are the 3 properties of partial order?**

**Answer:**

1. Reflexive: a <= a
2. Anti-symmetric: if a <= b and b <= a, then a = b
3. Transitive: if a <= b and b <= c, then a <= c



The above Hasse diagram shows the set of all subsets of the set {x, y, z}, ordered by inclusion (i.e., the subset relation). This is an example partial order (from the Wikipedia page on partial orders). For example, we see that the set {x, y} includes (is "greater than") the set {x}. However, the sets {x, y} and {x, z} do not include each other and are not equal: they are "incomparable".

The partial order depicted in this Hasse diagram has a distinguished greatest element, namely the set {x, y, z}. It also has a distinguished least element, namely the empty set.

There are rarely distinguished top and bottom elements in design spaces. There is usually no single best or worst design. Imagine this Hasse diagram with the top and bottom removed. That's what most design spaces look like.

**The Pareto Front**



This graph is used to represent the trade-offs in designs for two criteria. The ideal design will lie on the origin of the graph. For example, the ideal spaceship will consume no fuel and will weight 0kg. Practically though, the best design will fall on the Pareto front.

**Question: What are the axes on the graph that Bill Moggridge uses to contrast different design disciplines?**

**Answer:** Physical Design to Digital (Conceptual proposed by Danial Jackson) Design vs. Technical & Objective to Human & Subjective. (from May 6th scribe notes)

**Question: What are the core skills of a designer? (Provided by class)**

**Answer:**

* Creativity
* Analytical skills
* Wisdom
* Experience
* Communication
	+ **What experience did Fred Brooks have?**

**Answer:** In the 60’s/70’s, before there were version control systems, people would work separately on components and would try to make them fit together. However, this was not an ideal solution because it would cause overlap and merge conflicts. Fred Brooks experienced this in the IBM 360 project.

* Being able to identify relevant metrics

**Question: How do you generate new ideas? (Provided by class)**

**Answer:**

* Drugs (e.g. change morning coffee)
* Talk/ask people outside of your scope of design and regular life
* Modify existing ideas
* Tell people that it’s a question for another problem
* Research/read articles about a subject
* Lie until it becomes the truth
* Get ideas from movies/novels
* Complain (data gathering)
* Brainstorming
* Ask someone who is more knowledgeable (e.g. Derek Rayside talked to a reference librarian in preparation for this lecture)
* Sleep
* Experiment
* Physical exercise
* Take a bath
* Suit up Barney Stinson style (always wear a tie)
* Light physical activity (e.g. knitting, cook)
* Travel (far/near)
* Hobbies
* Use Yahoo! Answers
* Doodle

**Fun Question: Who was the Harvard professor who had his tenure revoked?**

**Answer:** Timothy Leary – he advocated that his students take acid.

**On Local and Distant Analogies**

**Local analogy:** comparing similar things (e.g. Mac was like Alto and Windows was like Mac).

**Distant analogy:** analogies between biological processes and genetic algorithm

**On Intense and Casual Mode Creativity**

Intense mode creativity is when you actually work on the problem and try to figure it out. Causal mode creativity is where you work on the problem indirectly. For causal mode creativity to work, you must have applied intense mode creativity previously. Sanders and Thagard discovered the claim that Alan Turing would run 50km.

Aside: Alan Kay has his own shower in the basement of Xerox PARC.

**Question: who was the first programmer?**

**Answer**: Ada Lovelace

**Question: Can creativity be done in a group?**

**Answer**: Yes. Examples:

* Brainstorming: unstructured
* Synetics: group facilitator knows the real problem and transforms it into another problem, gives it to a group to solve, and then change the solution to the real solution
* Six Hats: different hats (roles) that are assigned to each person in a group. Example: one person can have the “irrational” hat.
* Can’t have a person that says “it sucks” because it kills morale.

**Question: Are there systematic approach to finding new ideas?**
**Answer:** Yes, for example, morphological analysis.

**Question: If we were to relax the constraints, what would we get? Restaurants example
Answers:**

* Buffet
* Charge you for the time/space
* Bring your own food

**Weak Form and Strong Form**

Both “weak form” and “strong form” refer to generating new ideas by changing the technology.

**Weak Form:** change minor components in the design (e.g. change the ORM one is using).

**Strong Form:** change programming paradigms (e.g. students at Brown worked on introducing *continuation* in web technologies).

**Question: anyone can name a relational row store?**

**Answers:**

* Big Table - not suited for transactional model.
* Oracle, Sybase, MySQL, SQL Server

**Question: any other type of databases?**

**Answers:**

* Non-relational databases (NoSQL)
* At Google they use relational column stores for logging

Some benefits for using relational column stores [1]:

* They are more efficient when an aggregate needs to be computed over many rows but only for a small subset of all columns of data.
* They are more efficient when new values of a column are supplied for all rows at once.

**Sample Questions:**

1. Who is Ted Codd?

Answer: the inventor of relational databases

1. What property does the following equation infer? If A = B, and B = C, then A = C.

Answer: Transitivity

1. What is a necessary requirement for casual mode creativity to be effective?

Answer: Must have previously employed intense mode creativity.

1. What does it mean if a design is dominated? (in the context of a Pareto front)

Answer: There exists at least one other design that is strictly better than it on every criteria under consideration.

**References:**

[1] <http://en.wikipedia.org/wiki/Column-oriented_DBMS>

**Bibliography:**

All images and content was derived from the SE464 lecture slides.