The Hell, Michigan Round at the 2024 MIT Mystery Hunt

Exchange Gift for Gathering for Gardner 15 from Joe DeVincentis

I am a member of an MIT Mystery Hunt team. I have been on five different teams over the 25 years I have been participating, though to some degree every year's team is different since people come and go. In 2023 I was writing for the hunt in January 2024, my fourth time to do so, and I really feel like I did something special this year. Something that generated reactions like "That was amazing!" and "How could you possibly make that work?" and had a lot of people saying it was their favorite part of the hunt. I'm going to explain how I did it in this paper. Spoilers occur later on, with a warning and link if you want to try the puzzles yourself first.

About Puzzle Hunts

Puzzle Hunts are different from simple puzzles in that there are a group of puzzles meant to work together. The form of that interaction varies. A common style is the *metapuzzle*: Several individual puzzles lead to final answers that are usually words or short phrases. The metapuzzle is an incomplete puzzle that needs the answers from the other puzzles to complete it. Solving enough individual puzzles (called *feeder puzzles* because they *feed into* the metapuzzle) lets you solve the metapuzzle. It's usually possible to solve the metapuzzle missing some of the feeder answers; typically you need 60%-80% of the answers.

Sometimes metapuzzles use the answer words or their letters alone, with only some hints in the metapuzzle to help you figure out what to do with the answers; puzzlehunters call this a *pure meta*. Other times you may be given additional information in the metapuzzle itself, such as a grid, or you may get information along with confirming each feeder answer, which is used instead of or in addition to the feeder answers to solve the meta. These are called *shell metas*.

The Mystery Hunt has seen many variations on the general metapuzzle structure. Sometimes the feeder answers aren't words or short phrases. One round (a *round* refers to an entire group of puzzles and their metapuzzle(s); because of differing structures, some rounds include more than one metapuzzle) had every answer be a picture solvers had to find and verify with HQ (*headquarters*; the team running the hunt, also called *GC* or *Game Control* in some puzzle hunts). In another round, each answer was a physical object. In another, each answer was an emoji character. In yet another, each answer was a grid of letters and spaces arranged in a specific way. Sometimes individual feeder puzzles have multiple word-or-phrase answers you can find and confirm separately.

Other times it's the metapuzzles that work differently, especially when there are multiple metas. Sometimes you have to figure out which feeder answers belong to each meta. Possibly some feeder answers belong to multiple metas. Sometimes rounds build up in stages, with one metapuzzle using an

initial set of feeder answers, and a second metapuzzle using those answers again plus the answers of new feeder puzzles you are given after solving the first meta, potentially iterating more times.

About Our Mystery Hunt

There are no spoilers for the puzzles in the Hell, Michigan round in this section. The theme for the overall hunt is spoiled, but only to the degree that someone attending the opening skit would have seen (you can watch that opening skit at https://youtu.be/BwnsHRNyTns). The puzzle spoilers will begin in the next section (on the next page).

Puzzle hunts usually have themes and some sort of story that gives some reason for why participants are solving puzzles, though that often involves ludicrous leaps of logic and the assumption that everybody everywhere does puzzles all the time. You have to accept this, just like the suspension of disbelief necessary to enjoy TV shows and movies based on magic or other obviously unreal scenarios.

We chose to use a theme based on Greek mythology, and to some extent Roman as well, since the Romans worshiped many of the same gods by different names. Specifically, because of the popularity of Pluto in recent years, we chose to base our hunt on the demotion of Pluto to dwarf planet by the International Astronomical Union in 2006.

Our team had connections at Caltech, where Mike Brown is a professor; Mr. Brown was the astronomer most responsible for the IAU's decision, and wrote a 2010 memoir *How I Killed Pluto and Why It Had It Coming* about the event. We were able to get Mr. Brown to record a short video ranting about how he wanted to demote Pluto even further. During the opening, we played this video, then had a group of made-up astronomers vote to demote Pluto to "unimportant space debris."

Moments after the vote concludes, the stage is invaded by Persephone (Pluto's wife) and several other gods who are incensed that these *mortals* (they emphasize the word several times) have just killed Pluto. After some exposition about how the gods are the planets and vice versa, the gods present blame the entire group of mortals present at the event (the astronomers on stage and the solving teams watching in the auditorium) for Pluto's death, and banish them all to the underworld. Solvers are already presented with multiple tasks, all of them to be achieved by solving puzzles: Escape the underworld, restore Pluto to life, and appease the angry gods.

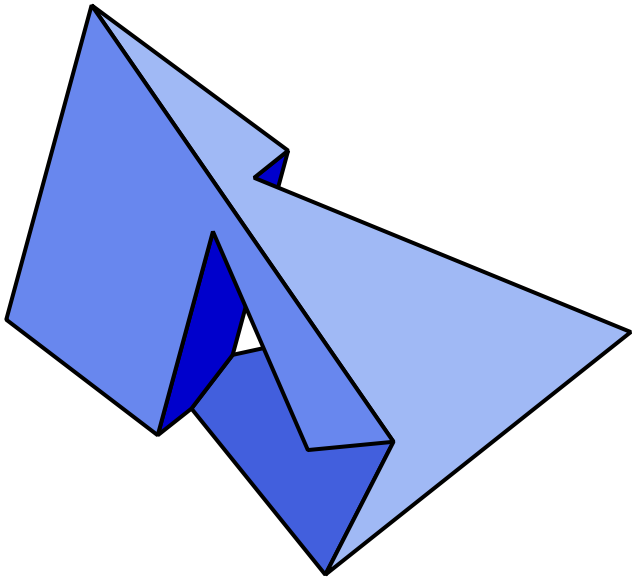
A Call for Unorthodox Round Concepts, and My Answer

A few months into constructing our hunt, our team leadership was worried that we would be letting down solvers who had become used to the more ingenious round designs that have become so frequent that solvers might expect to see some in every Mystery Hunt, and they put out a call for unorthodox round designs. This is where I stepped up with the idea for the Hell, Michigan round.

Last chance before spoilers begin. At the time of this writing, the puzzles are available by going to https://mythstoryhunt.world. Click "Public Access", and then from the Rounds menu click Hell, MI.

At some point this year, and possibly before you get a chance to read this, it will move into the hunt archive at https://puzzles.mit.edu/huntsbyyear.html under 2004.

I've always been fascinated by the Szilassi polyhedron. Some of you probably know it; it's a polyhedron with 7 faces, most of them concave, and each with 6 edges, sharing one edge with every other face. As you can see here, it has a hole through it.



My thinking was to make a round where puzzles don't work by themselves. We had such a round back in 2009, the Reverse Dimension, where each puzzle's long title described one of the first nine Doctors from *Doctor Who* or one of their companions. You had to pair up puzzle-halves for the Doctors and their corresponding companions to get puzzles you could actually solve, though the interaction between the halves worked differently for different pairs. This proved that giving solvers half-puzzles they have to pair up to solve was feasible.

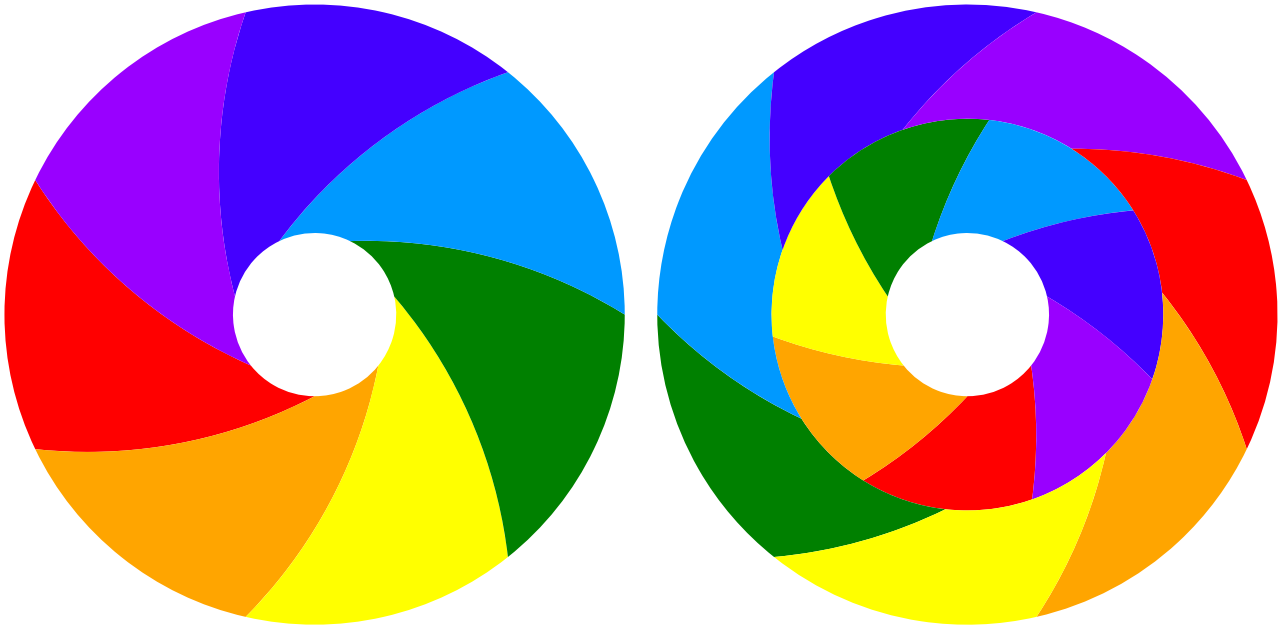
But my round was going to go further than that. Each of the seven puzzles was going to share an answer with *each* other puzzle, with some sort of interaction between the pair necessary to find the answer. That's C(7,2) = 21 answers. It seems like a lot (8 to 12 feeder answers is typical) but in rounds in the past where puzzles had multiple answers, we've had similar numbers. The *Legend of Zelda* Triforce round in the 2011 hunt featured 9 puzzles, each with 3 answers, for a total of 27 answers. (There were three metapuzzles, and each used one answer from each puzzle, but you had to figure out which.) The 2009 hunt had an Orbital Nexus round which was based on phases of the moon. This was represented within the hunt by each of the 8 puzzles in this round changing in subtle ways every 15 minutes, cycling through all four of its variations each hour. So you had to solve each puzzle four times with different data to get 32 answers. In contrast to these rounds, 21 answers wasn't too many.

Since I was putting puzzles on the faces and answers implicitly on the edges by virtue of them being shared by pairs of faces, logically the answer should extract letters from the vertices somehow. I decided the three answers meeting at each vertex would have only one letter common to all three and that letter would go there. Moreover, if possible, I'd have every two answers meeting at a vertex have at least two letters in common, to make it not too easy and give solvers an incentive to keep solving for more answers.

But then how are solvers to read the answer? A quick lookup told me there are 24 Hamiltonian cycles on the Szilassi. Combined with 14 starting points and two directions to read the answer, that's too many orders. Sure, if they have all the letters they can list all the possibilities via a program and scan the list of several hundred strings for something that looks sensible, but good metapuzzle design ensures the puzzle is solvable with a few answers missing. Turning, say, 8 of those letters into choices of two

letters each (because four answers touching 8 different vertices are missing) multiplies those choices by 256. A design concept for puzzle hunts is that random anagramming is bad. To give solvers so many letters without a stronger ordering concept than “it forms a Hamiltonian” is akin to random anagramming.

This led me to the next stage in this puzzle’s evolution. Instead of using the Szilassi polyhedron as the design concept, I would use the 7-colored torus. And I’d help them out a bit. By giving them the image (below left) with the 7 bands of color and referring to it as a doughnut, they’d know that images of the torus they can find online where the back side looks like the right image below are what they want. While there are still lots of Hamiltonians, there’s now one obvious one where the letters read around in a circle.



And looking up 7-colored torus in Google images gives plenty of images like this, including one from somebody selling pillows made in such a shape. This was a reasonable way to clue this information to solvers, who should have figured out every pair of puzzles shares an answer from the puzzles themselves, if they didn’t already get it from this image in the metapuzzle.

So I felt like I had a good metapuzzle mechanism. In most cases that’s enough to start writing an actual metapuzzle with actual feeder answers that you make up leading to a metapuzzle answer that you make up but with approval from the editors of the hunt. In this case, I wanted to do a bit more. Making each puzzle interact with every other puzzle in the round is a heavy burden, much more than feeder puzzles usually carry. So, in contrast to the usual meta authoring strategy where the feeder puzzles aren’t even considered until a metapuzzle to written and tested, in this case I wanted to plan out a feasible set of feeder puzzles to ensure I could do what I was proposing. I was pioneering here. I didn’t want to write and test a metapuzzle and then discover that I couldn’t figure out how to make the puzzles interact in all the ways necessary.

Feeder Interaction Design

In this stage I developed loose plans for what each puzzle should be and how they would interact. I wasn't actually writing the puzzles (with one exception), just considering how certain types of puzzles could work and how I could force them to interact.

The first puzzle I designed was one called Blanks. This puzzle was meant to be a clue that the puzzles needed to work together, that there wasn't possibly enough information here to solve the puzzle by itself. There would be just a set of blanks with five of them numbered like this:

```
_ _ _ _ _ _   _ _ _ _ _   _ _ _ _ _ _ _ _
    2     4       1       3       5
```

I had decided five distinct letters was the minimum any of my answers should have, just as two was the minimum any two answers meeting at a vertex should have in common. This puzzle would involve six phrases which fit on the blanks (one from each other puzzle) and would extract a five-letter answer from each. I wanted to minimize the strain on the phrases to spell good words, while minimizing the strain on the puzzles to spell out long and weird phrases (which there would be plenty of already apart from this puzzle) and this was the balancing point. The letters are well distributed among the three words and tests showed it was able to make a variety of different words with different vowel-consonant sequences.

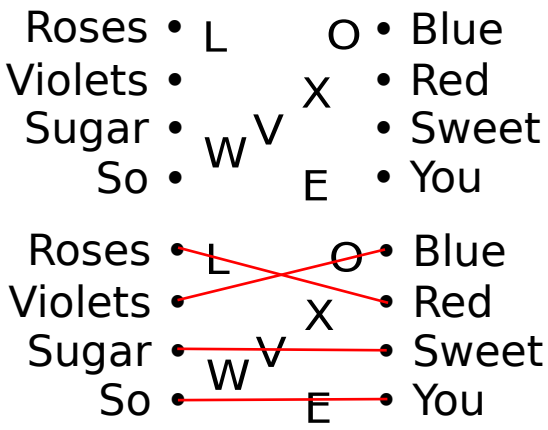
Next I decided to make three grid puzzles. A classic mechanism is to give solvers two grids of the same size to overlay. In this case, I'd write a 15x15 crossword, a 15x15 word search, and a 15x15 Akari puzzle. The last is a Japanese puzzle type sometimes called Light Up in English. You're to place lights in certain cells of a grid so as to illuminate all cells according to a number of restrictions. The light positions make such an obvious overlay that I felt a clue phrase telling solvers to overlay these puzzles was unnecessary.

But the word search and crossword should also overlay, and the mechanism there was less obvious. Perhaps the most obvious thing to do was overlay the grids and see where the same letters overlay on each other. But I didn't want to put that much more strain on the crossword and word search grids. So I decided to leave a message in unused letters of the word search to overlay it on the crossword and look through the Qs. And I'd just put enough Qs in the word search to match the answer for this pair, again to reduce the strain I was putting on the crossword puzzle. Word searches are less constrained and I felt sure I could make a word search with whatever else was needed after the other two grid puzzles were written. The Akari would be written first of the three, so that whoever was constructing the crossword knew where to put letters to spell out a message. And finally, the message among the unused word search letters explicitly referring to the crossword was another chance for solvers to figure out the puzzles had to work together, if they hadn't done that yet.

I also decided at this point that more unused letters in the word search could give the phrase to put on the blanks, the crossword could have the phrase as one of the clues, and the Akari, which hadn't been planned to have any letters in the grid at all at this point, could hide the phrase in flavor text. So this was four puzzles interacting in every combination. But I had three more to go, three which had to interact with each other *and* with each of these three.

I added one more puzzle type next because I knew it would come in handy.

This puzzle type didn't have a name until Mike Selinker called it Matchmaker in his book *Puzzlecraft*.



The basic puzzle type is simple: Match the items on one side with the items on the other, drawing lines to connect the corresponding dots.

This leads to some of the letters in the middle being crossed by lines. We might notice that the items on the right are in alphabetical order, while the ones on the left are in the order of the poem. Reading the letters crossed by the order of the left ends of the lines spelled out the answer LOVE. The letters not crossed by lines are red herrings and are not used.

But that's not the only way to read the answer from one of these. Sometimes the things in the middle are numbers, and you have to index them into the items on one side, perhaps reading them in the order of the connections on the other side. This wouldn't work for LOVE in this example since Sugar doesn't have any of those letters, and the letters on the right don't have a V. But suppose LOVE were replaced by 3, 4, 2, 1, respectively. Then, indexing the 3 into Red (and reading it first since it's on the first line from the left) gives a D. The other numbers in the same manner complete the word DEWY.

Sometimes it's the items that aren't crossed that are used. In this case there's nothing to index into, so the items generally have to be used directly. If we apply that to this puzzle, we get XW if reading the letters from top to bottom, or WX if reading from left to right. Neither of these looks useful, so we can conclude this puzzle doesn't work that way. But some Matchmaker puzzles do.

I had decided I would write the mother of all Matchmakers. There were going to be letters, numbers, and Roman numerals (in a different font to distinguish them from the letters) in the middle. Each line would cross one of each type, and some of each type would remain uncrossed, giving us 6 sets of symbols. The two sets of crossed numbers would each index into one side, read in the order of connections at the other side, using both choices of sides. The other four sets would read left to right, top to bottom, right to left, and bottom to top, a different direction for each set. The overloaded puzzle would give 6 messages, one for each other puzzle. One would be the phrase for the blanks, one would identify a crossword entry, one would identify a word search word, and I wasn't sure how to extract from the Akari.

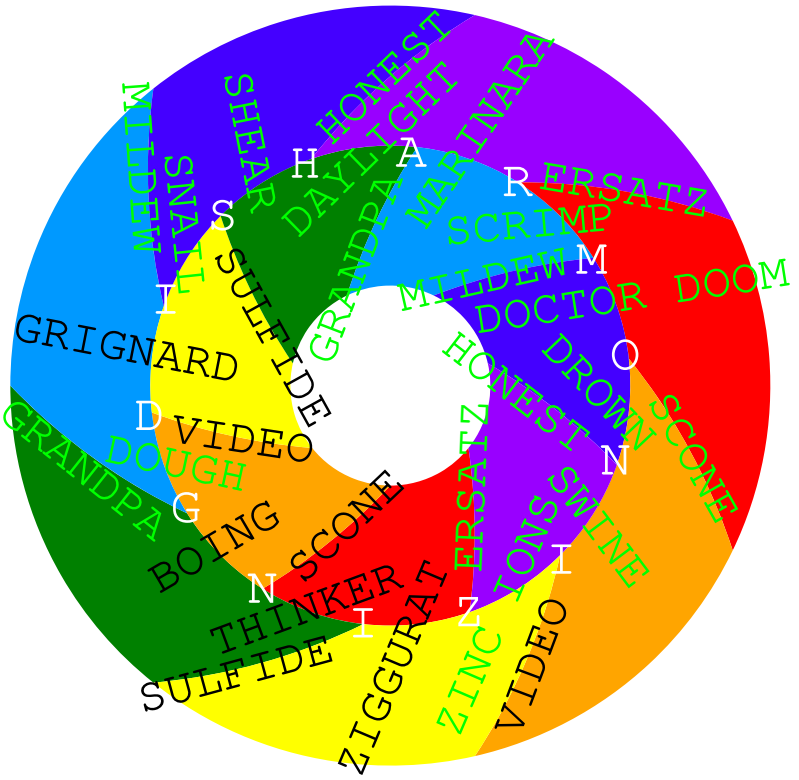
This was when I decided the Akari would have small letters in the cells. The letters on the cells with lights could spell a message to apply to some other puzzle, and one from the Matchmaker made of unused numbers would start SHIFT BY when indexed into the alphabet, and the rest would tell how to shift the letters of the Akari message within the alphabet to give a different message. I also decided the Matchmaker would be written last. It would be able to encode any messages of reasonable length, only limited to the two extractions from the clues being the same length.

One of the editors working with me suggested a technical puzzle ("something sciencey") and an interactive puzzle as good ways to fill in the last two puzzles needed to complete the set. I chose a

chemistry puzzle, because I felt I had enough knowledge to write it, and a text adventure, because it had been done before in Inform 7, so I knew it was feasible. Also, it was possible to hide easter eggs inside the game that messages from other puzzles might lead to, or give messages when completing tasks the game posed which would apply to other puzzles.

Writing and Testing the Metapuzzle

After deciding how the remaining interactions would work, including have the traditional long theme entries of the crossword identify an answer among the chemicals shown or implied in the chemistry puzzle, I started actually writing and testing the metapuzzle.



First was coming up with an answer. Some of the first answers I tried led to a problem completing feeder answer selection. Repeated letters in the answer forced so many feeder answers to contain that letter that the three answers meeting at some other vertices were forced to contain that letter. I didn't want that to happen; I wanted the letters to be unique is solvers had all three letters at the vertex.

I wrote a script to check possible answers for problems of this sort, as well as report where just two of the three answers at a vertex were forced to have the letter so I could be sure to avoid it in the third answer. With this sorted out, and a working meta answer chosen, I selected the other answers, choosing five-letter answers for the Blanks puzzle (shown here in orange) that I could make good phrases for, and chemistry-related words for some answers for the Chemistry puzzle (yellow) where I expected other puzzles to pick out aspects of the chemistry with a clue phrase.

Then we tested the metapuzzle. Testing rarely goes as smoothly as you hope, and this one was sure to be extra-tricky. Testers were spoiled on the idea that the feeder answers we gave them came from interactions from two puzzles and were each accepted as answers for both those puzzles. In one test, we gave them just the colors for each answer; the plan was that the puzzles would be labeled with their ROY G BIV colors and linked from those sections of the torus image we gave them. They got too distracted by there being multiple chemistry words in the yellow puzzle to actually focus on what they were supposed to be doing.

After a couple other configurations, we finally settled on this plan: Puzzles would not initially be associated with colors, but when solvers got all the answers for a particular color, the round page would

reveal to them which color it was associated with. And we tested this with 16 answers given (all the answers from three puzzles and one other answer) which seemed barely enough for them to get it, and in some other combinations, and it seemed to work.

But one editor pointed out that none of these testers tried to brute-force the solution. The meta answer DISHARMONIZING was supposed to be a pun, that Dis (another name for Pluto) had harmonized all the groups in Hades and stopped them from fighting little battles at all their borders. But disharmonizing is a real word and can be found in word lists. They wanted to know whether the answer could be brute-forced with fewer answers than we gave the testers. So I wrote a program to brute-force my own meta, and we found that if they just solved all the answers in two puzzles (so only knowing the color identities of those two puzzles) the answer just popped out, and other combinations suggested it might be gettable with even fewer answers. We didn't want teams to be able to bypass so much of the round, so we decided we had to scrap all this and start over with a new "non-Nutrimaticable" meta answer and new feeder answers. (Nutrimatic is a tool, beloved by some puzzlehunters, which lets you fill out partial answers with regular-expression-like syntax, sorting results based on commonality of phrases, with a huge corpus that includes all the text of Wikipedia.)

The new testers solved the new metapuzzle just as easily as the original using the final testing setup we had decided on, and we finally agreed this metapuzzle had passed testing.

At some point during this process, I was informed that, for puzzle story reasons, the round needed to be hosted in some American city. Our creative director, who was in this puzzle's discussion to try to figure out the art needs for this round, suggested Hell, Michigan. This is a real town, but in the real world a normal one, apart from being the butt of jokes about Hell freezing over. I agreed that it was reasonable to make a fictional portal there where we looked back into Hades and saw what was happening there.

Writing Feeder Puzzles

The normal result after a metapuzzle passes testing is that the answers become available to distribute to authors of feeder puzzles. My chief editor had been working with me on this puzzle and agreed that couldn't happen here. When he promoted the puzzle out of testing he immediately assigned all its answers to a placeholder puzzle, so I could write up more fully fleshed out placeholders for each of the seven feeder puzzles in the round, documenting all their required interactions, based on the detailed plans I had set up earlier. Once those were ready, my editor moved the appropriate answers into each puzzle and deleted his first placeholder.

I quickly wrote up the Blanks puzzle using the phrases I found while developing the final set of feeder answers, and then worked on the Akari. I had decided the interaction between the Akari and the Text Adventure was going to be that the Akari would have a theme of three central pillars labeled 1, 2, 3 (meaning they have lights adjacent to 1, 2, and 3 of their 4 sides, respectively) and in the text adventure solvers would encounter three numbered pillars with lights they could light up with text commands. When they matched the Akari solution, the game would give them the associated answer. This gave me a starting point for my Akari, so I wasn't simply writing any 15x15 Akari with no constraints, and I designed a puzzle I felt was of medium difficulty.

It was at this point that we gave both the Blanks and Akari to one of our factcheckers. Factchecking is so named because it often involves looking up bits of trivia that puzzles depend on, but it really means verifying that the puzzle works the way it should. For a logic puzzle like Akari, it means verifying the solution is valid and unique. For Blanks, it means making sure the phrases are spelled correctly and extract to the right answers. Usually we do this just before sending puzzles into test-solving. Because the next puzzles were going to depend on these puzzles working correctly, if there were any errors, we wanted to find them now. The Akari only used a Blanks phrase in its flavor text and would have been easily corrected, but the crossword was going to use one as a crossword clue, so the intended answer for this clue was also verified.

The chief editor I had been working with in the effort to get this round into the hunt was also a published crossword constructor (as opposed to myself being only an occasional amateur crossword constructor) and he volunteered to write the crossword with its odd variety of constraints. With that done and factchecked, I wrote the word search. I placed this last to construct of the three grid puzzles because I figured it had the most freedom.

I initially placed the letters in the grid needed for the two overlay extractions, and except for one Q saved for an unused-letter message, which I put near the end, I placed words in the word search to cover all those letters. I additionally added words to leave the right number of unused letters after the unused Q and the right total number of unused letters for the messages I was hiding. And then I put in the letters for the messages to fill the grid. I decided to make the word search a clued one, to give it a little more meat, and wrote some clues, and passed that on to factchecking.

There was a bit of a pause at this point, and when construction resumed, my chief editor had found a teammate Linus Hamilton eager to write a text adventure, and I'd found Alina Khankin eager to write a chemistry puzzle, and I explained to them the unusual constraints on each puzzle and the way they'd have to work together on one interaction.

Text adventure author Linus went all out, writing his game to present some classic puzzles like the towers of Hanoi and the river crossing but with constraints that made them not work. For instance, you were only given 20 moves to complete the towers with 5 discs, which normally takes 31 moves. You had to take advantage of aspects of the text adventure world or secrets revealed from other puzzles to cheat at each puzzle (in terms of its normal rules). Each puzzle was hosted by a particular historical figure, such as Genghis Khan at the towers. If you tried to cheat by putting the Hanoi discs somewhere other than the poles, you were prevented from doing so, and all-seeing Khan yelled at you in all caps THAT IS NOT A POLE or (for people) HE IS NOT A POLE, even if you were doing it in another room. But two of the historical figures, Copernicus and Chopin, were Polish, and hence Poles. Khan let you get away with using either or both of them to store discs for the puzzle, which made it possible to complete in the allowed moves. It took a lot longer to complete the text adventure than the other puzzles in this round, but it was also a lot of fun, and it was the single puzzle from the round most called out as solvers' favorite. Massive thanks, Linus!

Once I had the messages confirmed that the Matchmaker needed to send to these and all the other puzzles, I wrote the very overloaded Matchmaker. I had submitted another puzzle idea which could be implemented with matching, and not having any other good ideas for matching items to use in the

Matchmaker, after verifying I had enough items and the right letters to make it work, I decided to adapt this idea for Hell's Matchmaker. It involved works of fiction which uncannily predicted inventions of historical events, such as Morgan Robertson's *The Wreck of the Titan*, which tells of a shipwreck much like the Titanic, of the same size, on the same route, caused by hitting an iceberg, and with a similar name, but published 14 years before the Titanic sank. You had to match the works of fiction to the historical events, for which only the lengths of words and their initial letters and years were given.

Testing the Feeder Puzzles

Only once all the feeder puzzles were all constructed could we test any of them, because they all needed to be used together to get the answers. So we recruited some larger teams (compared with the usual pairs who would test individual feeder puzzles) with larger blocks of time. We also used this as an opportunity to test our post-production process (post-prod, for short).

The term post-production is borrowed from filmmaking, where it refers to the steps taken after scenes are filmed and audio is recorded, including editing, sound mixing, special effects, and the like. For our puzzles, it's the process between taking the puzzle the way it was presented to testers and reproducing it as a web page, consistent in style with our other puzzles. The team working on this had just gotten things ready to be able to post-prod puzzles, so these were used as guinea pigs, and the full-round test also tested our ability to host puzzles on a web site and to use the real answer checker as opposed to the one in our test-solving platform.

Fortunately, this went smoothly. Not only did the testers successfully solve the round, but they loved it. So did many solvers during the actual hunt, as they let us know through their cheerful and praising feedback. My success was built on making an ambitious goal, planning carefully, having the right insight to see how to make it work, and great teammates to help with planning, writing, testing, and factchecking.