Building a Hacker Space

Thorsten Haas <skytee@koeln.ccc.de>
Lars Weiler <pylon@ccc.de>
Jens Ohlig <jens@ccc.de>

Chaos Computer Club Cologne e. V.

August 17, 2007
Outline

1. Introduction
2. The Hacker Space Design Patterns Catalogue
3. Conclusion
Who we are

- Chaos Computer Club Cologne
- Founded in 1997
- around 42 members
- currently in Version 3.5 of our hacker space, operational in this location since 1999
What we want to tell you

- We want to share our knowledge of building our own hacker space
- We won’t give you a detailed manual
- We steal from computer science: Pick Design Patterns that work
- Your mileage may vary
Outline

1. Introduction

2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
     - Independence Patterns
     - Regularity Patterns
     - Conflict Resolution Patterns

3. Conclusion
The Infrastructure Pattern

Problem
You have a chicken-and-egg-problem: What should come first? Infrastructure or projects?
The Infrastructure Pattern

**Problem**
You have a chicken-and-egg-problem: What should come first? Infrastructure or projects?

**Implementation**
Make everything **infrastructure-driven**. Rooms, power, servers, connectivity, and other facilities come first. Once you have that, people will come up with the most amazing projects you didn’t think about in the first place.
The Community Pattern

Problem
How should your group communicate?
The Community Pattern

**Problem**
How should your group communicate?

**Implementation**
You are hackers, you know what to do. Stop slacking and set up a **mailing list**, a **wiki**, and an **IRC channel**. You will need all three.
The Critical Mass Pattern

**Problem**

You want to set up a hacker space in your city alone. You fail.
The Critical Mass Pattern

Problem
You want to set up a hacker space in your city alone. You fail.

Implementation
The rule of thumb is $2 + 2$. You need a partner to get the initial idea kicked off, making two of you. You need two more people in order to get real work done. Don’t start before you are at least four people. From this point it’s easy to recruit more people. Aim for ten people for a start.
The Strong Personalities Pattern

**Problem**

Nothing gets done. You all want the hacker space, but it’s so hard to get off your asses.
The Strong Personalities Pattern

**Problem**
Nothing gets done. You all want the hacker space, but it’s so hard to get off your asses.

**Implementation**
Look for strong personalities as members of your original group. You will need people with experience in building structures. Look for people who HAVE authority (and get respect), not for people who USE authority (and get laughed at).

T. Haas, L. Weiler, J. Ohlig
Outline

1 Introduction

2 The Hacker Space Design Patterns Catalogue
   Sustainability Patterns
   Independence Patterns
   Regularity Patterns
   Conflict Resolution Patterns

3 Conclusion
The Landlord and Neighbourhood Pattern

**Problem**

You have found the perfect hacker space, but the landlord seems to be weird. Also, the neighbours are picky.
The Landlord and Neighbourhood Pattern

**Problem**
You have found the perfect hacker space, but the landlord seems to be weird. Also, the neighbours are picky.

**Implementation**
Choose wisely. A benevolent, but uninterested landlord and cool neighbours can be the decisive reasons why the hacker space takes off or not. Not so cool neighbours may call the cops at 2 AM. Depending on your projects, this may be a serious problem.
The “It’s not a flat” Pattern

**Problem**
You need a space for meetings and as a lab, to store and work on materials for projects. In order to minimize rent or out of sympathy, you think it’s great when someone lives in your space. But somehow it doesn’t work, as you cannot use the lab anymore.
The “It’s not a flat” Pattern

Problem
You need a space for meetings and as a lab, to store and work on materials for projects. In order to minimize rent or out of sympathy, you think it’s great when someone lives in your space. But somehow it doesn’t work, as you cannot use the lab anymore.

Implementation
Guest are fine, but don’t let anyone live there. Kick them out if necessary.
The Séparée Pattern

**Problem**
You want to chill, discuss, or work in small groups. But the main room is occupied: There are simply too many people at your space. Or you want to smoke a cigarette at the space without disturbing non-smokers.
The Séparée Pattern

Problem

You want to chill, discuss, or work in small groups. But the main room is occupied: There are simply too many people at your space. Or you want to smoke a cigarette at the space without disturbing non-smokers.

Implementation

Look for a hacker space with smaller, separate rooms. Use curtains or doors to separate them from the main room. Separate rooms can also be used for smokers in a non-smoking hacker space.
The Kitchen Pattern

**Problem**

As a human being, you need food. As a hacker, you need caffeine and food at odd times.
The Kitchen Pattern

**Problem**
As a human being, you need food. As a hacker, you need caffeine and food at odd times.

**Implementation**
Have a kitchen at your space. Nothing brings people together like cooking together. Have fridges for Club-Mate. Selling soft-drinks will help you raise money for the rent. Invest in the single most important piece of hardware: a dishwasher. Have a freezer for pizzas and buy decent kitchen equipment. Show nerds how to cook real food.
The Coziness Pattern

**Problem**

All work and no play makes Jack a dull boy. There must be something else than only workstations and electronics.
The Coziness Pattern

**Problem**

All work and no play makes Jack a dull boy. There must be something else than only workstations and electronics.

**Implementation**

Bring in couches, sofas, comfortable chairs, tables, ashtrays, ambient light, stereo equipment, a projector, and video game consoles. Bringing in plants didn’t work for us.
The Shower Pattern

Problem
After long hacking sessions, you will start to smell funny. Also, guests to your space seem to neglect personal hygiene.
The Shower Pattern

Problem
After long hacking sessions, you will start to smell funny. Also, guests to your space seem to neglect personal hygiene.

Implementation
The discriminate hacker space has a bathroom with a shower. After a long hacking night you’ll have the best ideas while taking a shower. Guests from other hacker spaces may stay for several days. Ideally you will buy a washing machine to get rid of all the smelly towels.
The Membership Fees Pattern

**Problem**
You need to pay your rent and utilities. Larger projects need to be funded.
The Membership Fees Pattern

Problem
You need to pay your rent and utilities. Larger projects need to be funded.

Implementation
Collect fees regularly. Make no exceptions, ever. Choose an appropriate amount. Have discounts for students. Have at least three months of rent on your account, all the time, no exceptions. Elect a totalitarian treasurer.
The No Sponsoring Pattern

Problem

You think it’s a good idea to meet at a company that likes you or at a university where most of you study anyway.
The No Sponsoring Pattern

Problem
You think it’s a good idea to meet at a company that likes you or at a university where most of you study anyway.

Implementation
Never ever depend your space on external sponsors. Donations are great, but remember that companies can go bankrupt and you won’t be a student forever. Also, you will exclude high-school kids or people who don’t like the university culture. And no company, no matter how nice, will give away presents forever without asking for favours in return. That’s capitalism...
Outline

1. Introduction

2. The Hacker Space Design Patterns Catalogue
   Sustainability Patterns
   Independence Patterns
   Regularity Patterns
   Conflict Resolution Patterns

3. Conclusion
The Plenum Pattern

**Problem**

You want to resolve internal conflicts, exercise democratic decision-making, and discuss recent issues and future plans.

**Implementation**

Have a regular meeting with possibly all members. Have an agenda and set goals. Make people commit themselves to tasks. Write down minutes of the meeting and post them on a mailing list and/or Wiki. Go for the only date that works: once a week. Weird dates like "first full-moon after the third Friday" will never work. Likewise doesn’t bi-weekly or anything else.
The Plenum Pattern

**Problem**

You want to resolve internal conflicts, exercise democratic decision-making, and discuss recent issues and future plans.

**Implementation**

Have a regular meeting with possibly all members. Have an agenda and set goals. Make people commit themselves to tasks. Write down minutes of the meeting and post them on a mailing list and/or Wiki. Go for the only date that works: once a week. Weird dates like “first full-moon after the third Friday” will never work. Likewise doesn’t bi-weekly or anything else.
The OpenChaos Pattern

Problem

You want to draw in new people and provide an interface to the outside world.
The OpenChaos Pattern

Problem
You want to draw in new people and provide an interface to the outside world.

Implementation
Implementation: Have a monthly, public, and open lecture, talk or workshop. Invite interesting visitors to your regular meetings and don’t tell the weirdos about them.
The U23 Pattern

Problem

Your older members graduate from college or get married. Your space needs fresh blood.
The U23 Pattern

**Problem**

Your older members graduate from college or get married. Your space needs fresh blood.

**Implementation**

Recruit young people through a challenge you set up for them, in form of a course that spans several weeks. Overwhelm them with problems from hardware and software hacking and let them solve it in teams. Prepare for the challenge and tutor them, but give them room to experiment. After the team-building, retire and let the smartest of the young ones run the space.
Outline

1. Introduction

2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns

3. Conclusion
The Consensus Pattern

**Problem**

You need a group decision and want to make sure no one gets left behind.
The Consensus Pattern

**Problem**
You need a group decision and want to make sure no one gets left behind.

**Implementation**
Use the weekly plenum for discussion. Don’t take votes—**discuss until everyone agrees**. For some problems this pattern is the best.
The Democracy Pattern

Problem

You need to make a group decision. Discussion does not seem to lead you anywhere.
The Democracy Pattern

**Problem**
You need to make a group decision. Discussion does not seem to lead you anywhere.

**Implementation**
Use the weekly plenum for discussion. Do take votes—the strongest minority wins over the weaker minorities. For some problems this pattern is the best.
The Command Pattern

**Problem**

Nobody does the dishes. Your hacker space looks crappy. No one seems to care.
The Command Pattern

**Problem**
Nobody does the dishes. Your hacker space looks crappy. No one seems to care.

**Implementation**
*Order people* to do the dishes, take out the trash, keep the infrastructure up and running. *Yell, if necessary! Participate.* For some problems this pattern is the best.
The Responsibility Pattern

**Problem**
You volunteered for the task of running a critical piece of infrastructure, e.g. the mail server, but you feel the sudden urge to slack.
The Responsibility Pattern

**Problem**
You volunteered for the task of running a critical piece of infrastructure, e.g. the mail server, but you feel the sudden urge to slack.

**Implementation**
Just because volunteer work doesn’t get paid doesn’t mean it’s less important. Remember that you will directly hurt your friends and the hacker space. Take pride in your volunteer work. It will make you grow stronger as a person and is satisfying.
The Debate Culture Pattern

Problem

You are in the middle of your weekly plenum. Everybody’s yelling, nothing gets done.
The Debate Culture Pattern

**Problem**
You are in the middle of your weekly plenum. Everybody’s yelling, nothing gets done.

**Implementation**
Many geeks have very poor debate skills, the result of years of flame wars on the Net. Make people with actual social skills lead the discussion. Those with a background in real-life political work were best for our group. Learn from them. Learn not to interrupt others.
The Private Talk Pattern

Problem

Someone causes a problem that cannot be resolved in the group.
The Private Talk Pattern

**Problem**
Someone causes a problem that cannot be resolved in the group.

**Implementation**
Let some experienced member of your group talk to the trouble-maker in private. Listen to the person. Let them know how the group feels about the problem without exposing them in front of the group.
Conclusion

- There is no “golden way” building up a hacker space
- Based on experience there are a couple of patterns which might match
- Be creative! Try out your own way!