Building a Hacker Space

Jens Ohlig ‹jens@ccc.de› Lars Weiler ‹pylon@ccc.de›

24th Chaos Communication Congress

December 27, 2007

cc BY NC SA

Outline

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

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

The speakers

Jens

- Co-Founder of Chaos Computer Club Cologne (C4)
- Still active there
- CCC-activist for more than 15 years
- CCC-spokesman, board-member etc.

Pylon

- Co-Founder of Chaos Computer Club Düsseldorf (Chaosdorf)
- Now active in Cologne
- CCC-activist for more than eight years
- CCC-spokesman, board-member etc.

Chaos Computer Club Cologne

- Founded in 1997
- around 42 members
- currently in Version 3.5 of our hacker space, operational in that location since 1999
- All pictures in this presentation has been taken in the C4 hacker space

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

Thanks to the August 2007 hacker space tour

- A group of American hackers visited hacker spaces in Germany and Austria
- They wanted to know how our European hacker spaces work
- After the Camp they visited a couple of hacker spaces
- Every hacker space did a presentation about their history
- We created some Design Patterns

Design Patterns

- Historically used for urban planning
- Transfered for typical situations in software development
- Problem $\longrightarrow$ Implementation

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
- Your mileage may vary

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

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.

NOC with Server Racks

The Grace Hopper Pattern

Problem

Is now really the time to start your hacker space? Shouldn't you wait? Have you really thought of all the problems?

The Grace Hopper Pattern

Problem

Is now really the time to start your hacker space? Shouldn't you wait? Have you really thought of all the problems?

Implementation

Sure it is the time!

> *It's always easier to ask forgiveness than it is to get permission.*

(Grace Hopper, US Navy Rear Admiral and computer scientist)
It's important to start. Many problems you think of before will vanish as soon as you get started. **When in doubt, do it!**

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. Think about a **platform for discussion**, storage for **documentation** and **real-time communication**.

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 $\mathbf{2+2}$. 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).

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

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. As hackers you do not live the majority lifestyle—look for neighbours who are also weird and outside the majority.

The Roommate Anti-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 Roommate Anti-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 Kitchen

OpenBSD 3.7
I've a feeling we're not in Alberta anymore
www.openbsd.org
Preisliste
ANGST

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 Fnordcenter

Our Poor Plant Named “Egor”...

Sep. ’06

Dec. ’07

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 Bathroom (extract)

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 Sponsoring Anti-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 Sponsoring Anti-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. Meeting at a university will exclude high-school kids or people who don't like the university culture. No company, no matter how nice, will give away presents forever without asking for favours in return. That's capitalism...

Outline

1 Introduction
- Who we are
- Why this catalogue?

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

3 Conclusion
- This is not a cookbook
- Find your nearest Hacker Space

The Plenum Pattern

Problem

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

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 every other week or anything similar.

The Tuesday Pattern

Problem

Every weekday sucks. You will not find any day when every hacker can attend a meeting. Someone always has an appointment.

The Tuesday Pattern

Problem

Every weekday sucks. You will not find any day when every hacker can attend a meeting. Someone always has an appointment.

Implementation

Meet on Tuesday. Since all days are equally bad, just pick the Tuesday. End of discussion.

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

Have a **monthly, public, and open lecture, talk or workshop**. Announce it at your local time (no UTC, CEST, EST or something else). 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. Retire after the team-building and let the smartest of the young ones run the space.

The Sine Curve Pattern

Problem

You did everything right. You had some big events and a nice time in your shiny hacker space. But after some time the enthusiasm goes away and your projects are stagnating.

The Sine Curve Pattern

Problem

You did everything right. You had some big events and a nice time in your shiny hacker space. But after some time the enthusiasm goes away and your projects are stagnating.

Implementation

Peak enthusiasm at a hacker space has the form of a **sine curve with a cycle duration of four years**. Keep the hacker space running, even if the feel-good-factor is temporarily on holidays. Chances are your space will be awesome again in two years. **Don't give up!** Maybe an exciting new member will knock on your door tomorrow.

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

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! But always participate**.
For some problems this pattern is the best.

The `sudo leadership` Pattern

Problem

You started as a community of like-minded people, but suddenly you find yourself in a dictatorship run by a single hacker.

The `sudo leadership` Pattern

Problem

You started as a community of like-minded people, but suddenly you find yourself in a dictatorship run by a single hacker.

Implementation

Do not have ranks. **Use leadership temporarily**, like for projects and when you really need it. **Don't have a single root**.

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. When you realise that you really cannot do the job any more, **your last task is to hand it over**.

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 (e.g. student council) were best for our group. **Learn from them**. **Learn not to interrupt others**.

The Bikeshed Anti-Pattern

Problem

You suggest creating something new for your hacker space, like a bikeshed. But now everybody discuss about it's colour. No bikeshed will be built.

The Bikeshed Anti-Pattern

Problem

You suggest creating something new for your hacker space, like a bikeshed. But now everybody discuss about it's colour. No bikeshed will be built.

Implementation

That's a known problem. If you suggest something what everybody else in your hacker space can build, **they will take part in the discussion**. And if it's only the colour of the bikeshed, the design of the T-shirts, the Linux-distribution on the server, etc. Nerds tend to discuss trivial problems in epic detail, while more complex tasks will be ignored. **Identify pointless discussion** like these and just end them.

History of the Bikeshed-Problem

C. Northcote Parkinson wrote a book in the early 1960s, called "Parkinson's Law", which contains a lot of insight into the dynamics of management.

...

In the specific example involving the bike shed, the other vital component is an atomic power-plant, I guess that illustrates the age of the book. Parkinson shows how you can go into the board of directors and get approval for building a multi-million or even billion dollar atomic power plant, but if you want to build a bike shed you will be tangled up in endless discussions.

see `http://www.bikeshed.com/`

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.

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

The Old Hardware Pattern

Problem

You can't bring in shiny new hardware, as there is no space left.
Your space has become a hardware museum filled with junk.

The Old Hardware Pattern

Problem

You can't bring in shiny new hardware, as there is no space left. Your space has become a hardware museum filled with junk.

Implementation

Create a **pile/stack** where you put that old, unused hardware on. Let everybody take from it. Anything left within a while **should be thrown away**. But make sure you **announce that step** not only once, but at least three times with an escalation system.

(Sorry, in German only)

Bereitstellung von Hardware im Chaoslabor
§1 Das Chaoslabor ist ein Bereich vorbildlicher Ordnung und Sauberkeit, in dem der Chaos Computer Club Cologne e.V. seine Vereinstätigkeit ausübt.
§2 Unter Aufräumpersonal werden Personen verstanden, die sich um den Zustand des Clubraumes kümmern. Aufräumpersonal genießt Heldenstatus und Immunität gegenüber Anfeindungen von Besitzern nicht funktionierender Hardware.

...

see `http://wiki.koeln.ccc.de/index.php?title=Hacker_Space/Hardware-Gesetz`

A Stack of Old Hardware

The Key Pattern

Problem

You want the hacker space accessable all the time. You don not want to call somebody else during night to lock the hacker space when you leave.

The Key Pattern

Problem

You want the hacker space accessable all the time. You don not want to call somebody else during night to lock the hacker space when you leave.

Implementation

Hand out keys. Track who owns a key. **Have a good lock** so that nobody can copy the key without your permission. Collect a deposit for the key, so that the owner takes care for it. Or build a nice electronic locking system (with all cool things and all messy problems)...

The Club Mate Pattern

Problem

You need to raise funds. You want to stay up longer during night. You want to receive really good impressions without drugs.

The Club Mate Pattern

Problem

You need to raise funds. You want to stay up longer during night. You want to receive really good impressions without drugs.

Implementation

Buy at least one pallet of **Club-Mate** and sell it in your hacker space. You will realise the results very soon!

Golden Club-Mate

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

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!

- Question & Answer session @24C3: Day 1, 17:15, Workshoproom A

Outline

1. Introduction
   - Who we are
   - Why this catalogue?
2. The Hacker Space Design Patterns Catalogue
   - Sustainability Patterns
   - Independence Patterns
   - Regularity Patterns
   - Conflict Resolution Patterns
   - Creative Chaos Patterns
3. Conclusion
   - This is not a cookbook
   - Find your nearest Hacker Space

Hacker Spaces (not complete)

Germany, Austria, Switzerland:

- CCC-based → `http://www.ccc.de/regional/`
- Netzladen → `http;//www.netzladen.org/`
- Das Labor → `http://www.das-labor.org/`
- c-base → `http://www.c-base.org/`

USA:

- NYC Resistor → `http://www.nycresistor.com/`
- Seattle
- San Francisco
- L.A.

Australia:

- TheHacktory → `http://thehacktory.com/`