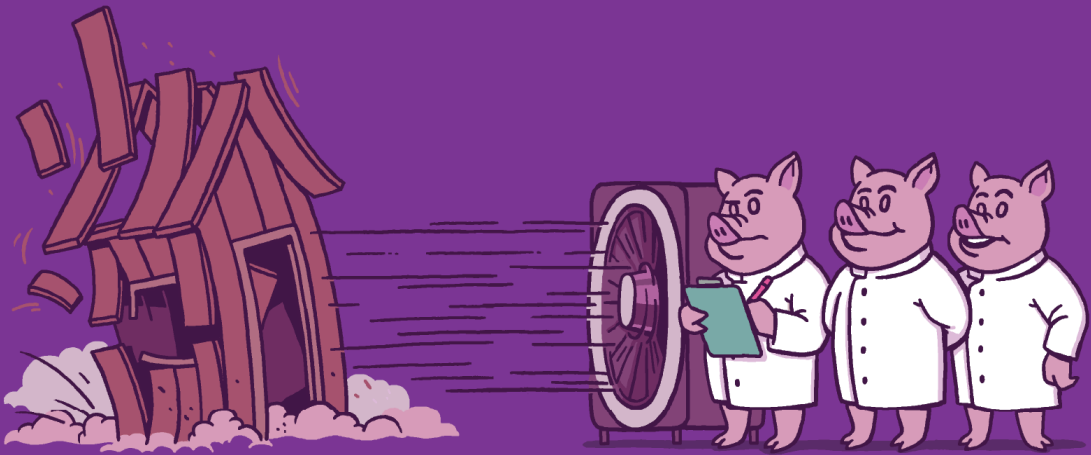


FIFTY QUICK IDEAS

TO IMPROVE YOUR

RETRO SPECTIVES



by Tom Roden and Ben Williams

Fifty Quick Ideas To Improve Your Retrospectives

Ben Williams and Tom Roden

This book is for sale at <http://leanpub.com/50quickretrospectives>

This version was published on 2015-10-29



Leanpub

This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

©2015 Neuri Consulting LLP

Tweet This Book!

Please help Ben Williams and Tom Roden by spreading the word about this book on [Twitter](#)!

The suggested hashtag for this book is [#50quickideas](#).

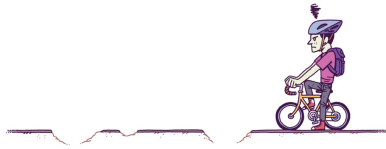
Find out what other people are saying about the book by clicking on this link to search for this hashtag on Twitter:

<https://twitter.com/search?q=#50quickideas>

Contents

Count your interruptions	1
Create a business case	5
Do a face-to-face team 360	9
Don't just say 'we can't'	13
Don't let facilitators solve problems	17
Get a room	21

Count your interruptions



We all get distracted at work. If we had a pound for every time one of us got distracted while in the middle of something, then – well maybe that’s an experiment to try out!

Open-plan offices are very popular these days. They tend to be noisy places and agile software development teams tend to create some pretty vocal environments. So when a number of teams are working closely in a department there are a fair few decibels and a lot of interaction. Context switching is taxing (literally). Moving between tasks and getting familiar with the new context takes time and grey matter, particularly when we are deeply immersed in knowledge work.

In the book *Peopleware*, Timothy Lister and Tom DeMarco talk about the heavy cost of open-plan environments in terms of the sheer amount of distraction. Since they wrote that book, the number of distractions and the different types of interruptions have increased. Just think about the explosion of social media and the increased tendency of applications to make themselves known, such as Outlook flashing each email onto your screen as it arrives. Lister and DeMarco also describe a state that psychologists call ‘flow’, a state where a person is intensely focused on a task, time seems to fly and people can just burn through work. This state of being ‘in the zone’ (different to the state felt after a few pints of beer when one, usually misguidedly, feels unbeatable at pool) can take

up to 15 minutes to reach. The research into flow also suggests that work needs to be challenging to take us into this state of intense concentration. A fair number of tasks in software, whether design, development or testing, are challenging. So distractions can cost a lot of wasted time, and therefore money.

The cost of distractions is made up of both the length of time we get distracted for and the absolute number of distractions we suffer. If you are getting distracted several times a day, or worse, several times an hour, the cost of losing where you were and having to immerse yourself back in the task again is very high.

Start to count your interruptions and distractions – it can be quite revealing, even shocking. Every time a member of the team is interrupted or distracted by something, count it. A person could do this on their own, but it works very well as a team activity. Count all the interruptions and distractions for a given period of time. This will give you some data to start analysing and using as part of the input to your retrospectives.

Key benefits

Just through measuring the interruptions, teams start to become really aware of quite how many times they get distracted. Teams start thinking about the associated cost, particularly when they are deeply engrossed in a piece of work.

This approach provides a visible way of measuring waste incurred through interruptions. It's a light-hearted approach too, which gives teams a better chance of sticking in teams' daily processes.

Collection provides a good source of data for input into retrospectives and for experiments with improvement ideas, especially if you break the interruptions down into types as described below.

Another great use for this data that we've found is that it helps teams to establish working patterns for quiet or interruptible time.

That can be really helpful in a large department or project, working closely with many other teams. Having conversations with others about the best times for avoiding interruptions can lead to agreements on quiet time and lead to happier, more productive teams, synchronising their periods for immersion in the deep flow of working on challenging tasks.

How to make it work

It is hard to quantify losses of concentration in absolute terms, so there is possibly more value in the measure as a trend in the number of distractions. Once a baseline number of interruptions has been established, the team can then experiment with how to reduce them.

It is not easy to calculate an exact time lost per interruption. However, if we make an assumption that it takes 10 minutes (instead of 15, to err on the lower side) to settle into a task, we can make a fairly simple calculation of the cost we incur. For example, in a team of eight people, if each person is interrupted just 6 times a day, then that is 8 hours gone. A whole person, or 12.5% of the team's capacity, is wasted every single day. That is just by being interrupted, and does not include the time spent on the interruption itself.

The other key thing to decide on is how to record the data and to what level of detail. Try to keep the data visible and make the process light-hearted, so people remember to do it and are motivated. The simplest way is just a tally system on a whiteboard. Some teams favour more elaborate record-keeping, collecting other pieces of information like time of day, type of interruption, duration, what they were doing at the time and even the response they gave (an answer, pointing to documentation, teaching, 'wrong number').

Some physical action performed when a person moves between tasks can work well. A colleague of ours suggested that to make it more engaging for the team, they should throw pieces of Lego into a box using bricks of different colours for different categories

of interruption. For example, red for a telephone call, white when a person comes to the team, blue when an email arrives, etc.

On a cautionary note, it's important to make sure people realise this counting interruptions is not about stopping communication between colleagues, or making it unacceptable for anyone to ask for help from another team. It is about identifying the unnecessary stuff and maybe rethinking the way in which we use the plethora of tools at our disposal. From a team perspective, it can still be desirable to put aside periods of time to work without any interruption at all, in order to enable members to focus on challenging tasks.

Create a business case



Teams can have some great ideas about improving their processes, some of which might involve significant investment. It is common however to see their grand plans quashed when it comes to getting permission or financial backing. Improvement ideas can sound as requests for new toys and observations can appear as complaints.

In addition, a team that presents an improvement idea without framing it within a business case is less likely to be funded by the people who control the budgets. This might either be because the business stakeholders don't understand the value proposition or they just might not take it seriously, because it is not written in their language.

If the team is not given feedback as to why their idea was not funded, and have further requests turned down, they may think 'they will never go for that' and not even mention the next big idea. If this happens frequently, potentially great ideas will be dismissed in favour of those that require much less investment but that are also much less likely to provide equivalent results.

If an improvement idea needs investment of any significant amount, then it is likely to be necessary to convince an internal stakeholder, quite possibly from outside the IT department, that it is a sound investment. If teams present improvement ideas with a business case they will have a much higher success rate.

Instead of asking for open-ended time and investment, present a likely return within a reasonable time period, with an appropriate amount of associated risk.

Key benefits

Just going to the effort of quantifying the opportunity might be enough. Often the potential benefit far outstrips any cost and there is no need to go any further in creating a business case.

Going beyond quantification and producing a full business case will make it easier to get funding for improvement ideas. This is for several reasons:

- Business cases prompt people to offer a rational response to a request by presenting information in a format that is easy to understand.
- They information is in the format that business people can use to make investment decisions.
- Finally and most importantly, they present ideas as likely outcomes: If I invest x, then I get y.

Business cases are versatile; they can be used throughout the organisation (and probably already are). We think that teams can use them for improving their way of working in the same way that whole departments can use them to fund a large change initiative. After all, businesses are built around the successful use of this technique.

How to make it work

At one of our financial services clients, the team calculated the break-even point in time savings for investing in automating regression testing. The department had an application with an ever-growing manual regression test pack that was taking around 100

person days to run per release. The team estimated the total effort to automate the testing at about 240 person days. It was then simple to calculate the break-even point as 3 regression pack runs, so the team put this business case in front of the head of department who immediately chose to fund the improvement opportunity.

This simple calculation did not even take into account the opportunity to reduce risk through task automation. When compiling a business case, consider including the risk angle. The decrease in risk can be significantly more important than the explicit value.

Another way to make this work well is to provide several options. Having multiple options to compare and contrast not only increases the chance of success, but may itself also increase the likelihood of a decision to select one of them. As Alan Weiss explains in *Million Dollar Consulting* regarding selling choices to clients, ‘always provide options – a choice of yeses’.

In order to compare investment opportunities we need to relate them back to the same currency. For example, asking someone to choose between something that will take about a week and an alternative that costs \$10,000 is unhelpful.

In a business with limited investment capital, different parts of an organisation compete for the same capital. Teams who can describe an opportunity in a concise business case will find win the lion’s share of investment decisions.

A word of warning: as with any other business case, the assumptions included when formulating the plan will be tested. If a team makes unfounded assumptions that exaggerate their business case, then they may find future business cases reviewed with a much higher diligence. Remember the boy who cried wolf.

We’ve known teams who say that they cannot be certain about the likely cost of investment or the likely benefits to be realised; the result is that they do nothing. Instead we encourage teams to be brave. A plan is not a contract, it is an improvement idea and its

primary purpose is to aid a conversation. By creating a business case, teams make the assumptions explicit. They have the chance to openly discuss our assumptions with colleagues and those in charge of the money.

With investment cases, teams are asking their internal stakeholders to use their business experience to judge proposals. Once people start framing improvement opportunities in terms of costs and benefits, they are often surprised at how engaged their partners become in selecting investment opportunities.

Do a face-to-face team 360



Giving personal feedback is often overlooked during retrospectives. Teams tend to focus on process or technical issues while ignoring human ones. This means that opportunities for personal improvement are lost or left to an annual appraisal process where feedback could come second-hand through a line manager.

Giving and receiving personal feedback can be a very difficult thing. For feedback to be effective, the recipient has to trust that it is given with good intention, as a constructive observation. The giver of feedback also has to trust that the receiver will receive it in that manner. This level of trust can be hard to achieve and for some just does not come naturally at all.

Trust cannot be demanded or forced, it is built over time. Often a good level of trust may not be reached because team members do not have an environment where they feel safe enough to expose their vulnerabilities. It is much easier to talk safely about processes and technical improvements because these are not personal in nature, nor likely to drive strong emotional responses and cause conflict.

A useful technique to encourage personal improvement is team 360 feedback. Encourage everyone to give feedback on everyone else, each person in turn becomes the subject of feedback. Going round the room, each of the other team members gives feedback while the receiving person notes down whatever they wish to take away and

work on. Make sure to time-box the session so that everyone gets enough time and no-one gets rushed at the end or misses her turn for feedback.

The main aim of this technique is to provide richer and more rounded feedback by the inclusion of many different perspectives. The key to this particular type of 360 feedback is the face-to-face aspect. Make it expected and desirable to give feedback directly and in front of the rest of the team, as part of the retrospective session.

Key benefits

Because the expected norm is to provide feedback, opting out of doing it becomes very hard. The point isn't to make people feel socially awkward, it is to overcome the initial discomfort, enabling personal feedback to become acceptable and forge better understanding. Ultimately it allows people to give feedback they would not have been comfortable initiating otherwise.

A significant benefit of this one is that, over a few runs, it becomes much less awkward to say what needs to be said. Sometimes the pursuit of continuous improvement brings with it the need to surface some sensitive issues. This idea helps make those conversations possible and hopefully lets criticism be given and taken in a constructive manner. If the each team member expects to give and receive direct and specific feedback, avoiding it becomes an effort.

The strength that the team gains from this form of feedback is immense. It may feel strange and uncomfortable to start with, but when it is understood that the purpose of all the feedback is to make the team better, these feelings quickly fall away. Do not forget the positive feedback that team members receive. This appreciation strengthens the team bond and ultimately helps each member understand where on the field they play best for the team.

When the 360 feedback happens regularly, it reduces the potential for a pressure-cooker explosion, where unsaid feedback can leave

feelings simmering away for weeks and months, which can be very damaging to a team. The face-to-face nature of this feedback breeds trust. As a member of a team receiving this type of feedback, I know where I am, all my team members have been honest with me and it is now up to me what I do with that feedback.

How to make it work

A technique that works well is for team members to propose one strength, one example where they felt the individual did well, and one specific area for improvement. Other variations of this can be for each participant to score themselves against criteria and the team to mediate their scoring with its consensus being revealed at the end. You will hear things like, 'John, I don't think that you are a 4 in that, Sally has herself as a 4 in that and I don't think you are quite at Sally's level'. In this technique, the criteria make the feedback more objective. A possible drawback is that a criterion is too specific and team members could be restricted on the scope of the feedback that they give. In addition, if in using this technique, it turns into a tick-box exercise with no discussion of the reasoning, most of the value is lost.

For a bonus point, try including the coach, facilitator or Scrum master, provide them with feedback and receive a different perspective on the team.

Here are some other tips for giving and receiving feedback:

- The only acceptable answer to feedback is 'thank you'. Don't defend your position, this may prevent further feedback.
- Saying 'thank you' is very powerful. Appreciation will grow trust and strengthen the team bond.
- Use concrete examples. This makes feedback more impactful, more genuine and easier to understand and cement in the mind.

- After the retrospective, refer to someone independent for their opinion on controversial feedback.
- The subject of the feedback might open with an honest assessment of themselves. People tend to be harder on themselves than others, but being self-critical makes feedback from the rest of the team more easy to handle.
- Provide feedback in a sandwich format: two slices of appreciation around an improvement filling.
- As team members get more trusting with each other, use Atkins feedback (no bread, just the meat in the middle).
- Similarly, make the positive feedback related to the improvement idea 'I like this.... you could improve it by.....'.

Don't just say 'we can't'



A statement of fact such as 'we can't do it' tends to close down any further conversation. Often, this is not said deliberately to derail a process or meeting, rather the team or person has said it for one of two reasons:

- They are limited by their experience and knowledge.
- They might have made a judgement in their mind that the cost benefit analysis is not favourable.

In both cases, the outcome is verbalised as 'we can't'.

With infinite resources anything is possible. If we had access to the brightest minds in the world and enough time, we could achieve anything. Saying 'we can't' and having unlimited resources are two extreme ends of the spectrum and things are not usually binary if we stop to consider them in more depth.

To exacerbate the problem, the other team members rarely argue against such a statement. This is because the person most likely to have an opinion on the feasibility of an idea is the most experienced one. Alternatively, the most dominant person in a team often makes such strong statements. In both cases, the others will be discouraged from challenging their judgement.

The outcome is that we just dismiss some solutions out of hand without discussing their underlying assumptions, and ultimately

we miss out on potential improvements. So instead of just saying 'we can't', spend some time exploring why. The results can be very revealing!

Explore the reasons for the initial reaction. Challenge the underlying, unspoken, assumptions. Quite often we find that teams unearth one or two of these underlying assumptions that they fundamentally disagree on. At this point, we have achieved a much more focused discussion.

Key benefits

Refusing to stop at 'we can't' means we don't prematurely rule potential improvements out due to false or unexplored assumptions and miss out on opportunities to explore improvement ideas.

Also significant is that asking 'why' helps the team to think of more possibilities and start to think more factually about what is involved in decision-making and what makes an idea a good investment of time (effort) and money. This will make them more capable of independent decision-making and problem-solving.

It also spreads knowledge across the team of the pros and cons of possible improvement ideas, bringing team alignment and hopefully autonomy.

Lastly we have applied collective reasoning: we have used the creativity and experience of the collective team to refine an improvement idea. This in itself improves the way the team functions.

How to make it work

Find some way of not allowing 'we can't' to be a satisfactory final statement. A retrospective facilitator such as a scrum master can specifically look out for these 'statements of fact', but a better idea

is to make it one of your team principles not to use the phrase 'we can't'.

It is quite often handy to ask these questions:

- If we had infinite time and money, would the response still be 'we can't'?
- Ask 'based on what?'
- Repeatedly ask 'why?'

As a start, imagine that you had infinite time and money – would the answer still be the same? We've found that posing the infinite resource question leads people to unravel the inbuilt assumptions about what makes the proposal difficult.

Another way to open up this dialogue is to ask 'based on what?' This question makes people analyse their thoughts, and hopefully challenge some of their own assumptions. Get team members to present their assumptions to the rest of the team, put them up on a board, prioritise them and then discuss whether they're valid and what the cost of changing them might be.

A third useful question is 'why'. Its use, increasingly popular in software delivery circles, is described by Taiichi Ohno in *Toyota Production System: Beyond Large-Scale Production*. Toyota used a succession of 'whys' to drill down to the root cause when they were trying to solve problems. They found that five whys were probably sufficient to drive out the true root cause rather than the immediately obvious but possible superficial cause. We can apply a similar technique, one often used by children to annoy their parents when they tell them 'no'. Keep asking 'why' until we flush out all the real reasons why people think that something is going to be difficult or impossible.

Funnily enough, it turns out that experts are human after all. They are constrained by their past experiences, not just the technology or

domain, but also their past successes and failures. After exploring assumptions thoroughly, sometimes you hear something like 'I don't know, it's just always been that way', which is a red flag for a 'please challenge me' question. Remember, their reasons for rejecting something based on previous experience may no longer be valid, as technology and the tools are changing all the time.

Don't let facilitators solve problems



Whether it is right or wrong, it is a matter of fact, most facilitators have been team managers at some point or another. Moreover, as a manager, their primary responsibility was solving problems for those they managed. This is a hard dynamic for many facilitators who might deny they ever solve problems but then eventually fess up.

Most managers care passionately about the teams they work with and can often see a way to resolve the team's immediate problems. So they think it helpful to step in and suggest a solution for the team to execute, or worse, take the responsibility for fixing the team's problem themselves.

The drawback is that, although it is tempting for the facilitator to fix the problems and in the short term, the facilitator is actually denying the team the opportunity to develop the skills to solve problems for themselves in the longer term. 'Hold on, my Scrum master's job is to remove impediments to the team' we hear you say. However, this creates a dependency on the facilitator, which might well get stronger rather than weaker over time.

Try to work by the principle that everything facilitators do for a team should reduce the team's reliance on facilitation a little and make their role a little more redundant (remember that when you

make your role redundant you are more likely to get promoted than fired).

There are two recurring failure modes that we see in this situation and as facilitators we must avoid.

First, as facilitators, we must resist jumping in. We might think we are acting in the best interests of the team by coming up with a solution to a problem they have identified, but in reality we deny them the chance to solve the problem for themselves.

When a facilitator describes a solution rather than a need, it is likely that any resulting outcome will be sub-optimal. Of the many reasons for this, not using all the skills and brain power of all the members of the retrospective is wasteful. Also there is a lack of ownership when a team are handed a solution rather than presented with a problem. The upshot is a team who might not appreciate why they are doing what they are doing while also missing out on an opportunity to improve their problem-solving skills.

The second, and arguably worse, failure mode is when the facilitator takes responsibility to fix the problem. In this situation, a team identifies a problem in a retrospective which the facilitator steps in to take away and fix, because they know the systems, organisation, people or solutions of the past. In this case, teams have many of the same problems as before but with the additional disadvantage that team members may not even know how the problem has been solved.

When the facilitator solves the problem, the next time the team face a similar problem, they will not know how to go about starting to rectify the situation. Instead, they will either have to find the facilitator or, even worse, wait until their next retrospective to raise this latest problem.

Key benefits

This idea means playing the long game. Used successfully to remove the reliance on the facilitator, it will provide greater long-term speed and throughput for your team.

By developing the problem-solving techniques that teams naturally have and applying them to solving organisational problems, team members will become more empowered. They will feel more ownership of a solution to a problem if they have designed it. This ownership will translate to a greater success rate.

This empowerment will bring a shift of focus within teams to fixing problems rather than sitting on them waiting for help, or just working round them. This agitation will start to change the dynamic and culture within an organisation as problems become things to solve and not immovable obstacles.

How to make it work

As a facilitator, back off. Even if you see an opportunity to solve a problem in a more efficient way, remember that is a short-term efficiency while the long-term effectiveness of the team depends on the team learning how to fix problems themselves.

Make sure that the team get to the root cause of the problem. Using techniques like the '5 Whys' and Ishikawa's fishbone model, widely written about in quality control literature, will drive the team to find the root cause of their problems. In this way, solutions that address the root cause become much more obvious than when teams only deal with the superficial problem first identified.

Use open questions in retrospectives to help the team drive their own solutions. Ask focusing questions like 'What possible courses of action do we have and which can you control or influence?' This should lead them to think about the range of options available.

Write these suggestions down, even if they seem impossible, as the act of displaying them will likely prompt exploration of the underlying assumptions and further derivatives of the original idea.

Finally, no one likes to see people struggle with a problem at length. If you as a facilitator must offer your insight, first, ask permission. Once you have the agreement of the team tell the team explicitly that you are taking off your 'facilitator hat' and that you are contributing as a team member. Once you have presented your idea as an option, don your facilitator hat again without trying to influence the team further.

This doesn't mean letting the team drive off the cliff. You can always challenge an idea, but remember that every time you step in, you reduce team's independence a bit.

Get a room



The prime directive for retrospectives described by Norman Kerth in his book *Project Retrospectives: A Handbook for Team Reviews*, emphasises the key point that safety is paramount for success. Airing your dirty laundry in public is never easy, so the degree of safety a team feels is directly affected by the environment they conduct retrospectives in.

If the team thinks they are being judged by co-workers, or by those who might determine their pay, career progression and ultimately success, then they are not likely to feel safe. Under these circumstances any retrospective is likely to be superficial at best, pointless at worst.

For a team to truly improve, they need to show vulnerability, and to do this, they need to feel safe. Part of feeling safe is having privacy.

The easiest way of having privacy is to get a room. This may seem simple and obvious, but its effects are very powerful. The team needs to feel they can talk about absolutely anything, safe in the knowledge that 'it will go no further than these four walls'.

Key benefits

Getting a room for your retrospective session pays back dividends. It ensures that you are getting the most from your investment of

time by encouraging open and honest dialogue about the things that matter, as opposed to ignoring the big issues.

Having a room will change the team's attitude to retrospectives. Not having the trappings of everyday working life in front of them emphasises the importance of the time spent in the room. Making the investment in the session by going to a room rather than just staying in the team space sets the tone for investment in continuous improvement itself.

A further significant benefit to getting a room is that it removes many other distractions that plague the modern workplace. Moving to a room reduces the amount of disruption from emails, any other kind of work or people approaching the team. One way you can further this is to agree to ban various items from the room such as phones, laptops etc.

How to make it work

If the team has a permanent room to themselves, then lucky them – most teams we have worked with don't. If not, then book a meeting room on a recurring basis.

When are picking a room, choose what works best for the team. If they are not happy with the overhead of going upstairs to the meeting suite on the ninth floor, make sure that the booking is close to the team's work area. Alternatively, as mentioned in the section *Take it outside*, it may be desirable to get completely away from the workspace.

Make sure you can't be overheard or disturb anyone else, so you can laugh and let off steam without inhibiting the outcome of the retrospective. Ben worked with a team who wrote software for electronic trading systems. In a retrospective session, they had a colleague come looking for a member of the team to answer a question to allow them to continue their work. The effect of that

interruption, the removal of one of the team members and the subsequent break killed the retrospective. The team struggled to pick up where they left off. At the next retrospective, one of the team members came to the session with an immediate improvement idea. They suggested that they put a sign on the door 'Retrospective in Progress' in order that people looking for a member of their team realise that it is a meeting that they can't interrupt. The rest of the team immediately agreed with the proposal, adopting it into their ongoing way of working.

The layout of the room is important too. Ideally the room layout should be an open space with seats in a circle, as Esther Derby and Diana Larsen point out in *Agile Retrospectives: Making Good Teams Great*. It may seem superficial but the physical barrier that a table brings also introduces a psychological barrier between members of the team.

Don't skimp on space and whiteboards to display artefacts from the team and the mandatory sticky note invasion, but also to get creative, draw diagrams, pictures, grids, emotions – whatever comes up! If this is not possible then try to get a room with enough flip charts. Though be warned – super sticky Post-it notes are easily rearranged on whiteboards, but it's much harder if you've stuck them on flip charts.

We worked in a building in London that had meeting rooms where two sides of each room were floor-to-ceiling windows. It was up a few floors, so a nice environment for retrospectives, just not very practical. In a retrospective that Ben was facilitating the team ran out of room on the whiteboards, but they simply kept going onto the windows. The point here is to adapt your environment to your needs, or adapt your needs to your environment.

Other teams in the same building commandeered empty offices to hold their retrospectives. They felt that although the rooms were small, it was a price worth paying for privacy.

Although it's very important that people can visualise the topics of

conversation, it's also necessary to retain the level of safety built in to the retrospective. As with good test practice, we should tear down our retrospective environment, by removing all the information we put on the walls and disposing of it safely.