momento

Appendix

Graduation thesis
Stijn Bakker
Table of contents
<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Detailed overview of research methodology.</td>
<td>4</td>
</tr>
<tr>
<td>B</td>
<td>Interview results</td>
<td>6</td>
</tr>
<tr>
<td>C</td>
<td>Codebook</td>
<td>14</td>
</tr>
<tr>
<td>D</td>
<td>Overview of meetings</td>
<td>18</td>
</tr>
<tr>
<td>E</td>
<td>Interventions overview</td>
<td>19</td>
</tr>
<tr>
<td>F</td>
<td>Interventions product team</td>
<td>20</td>
</tr>
<tr>
<td>G</td>
<td>Interventions organisational context</td>
<td>22</td>
</tr>
<tr>
<td>H</td>
<td>Discussions</td>
<td>27</td>
</tr>
<tr>
<td>I</td>
<td>Expert interview transcript</td>
<td>29</td>
</tr>
<tr>
<td>J</td>
<td>Excerpts from culture books and employee hand-books</td>
<td>33</td>
</tr>
<tr>
<td>K</td>
<td>Momento - mini-booklet</td>
<td>48</td>
</tr>
<tr>
<td>L</td>
<td>Momento - Little book of ODS</td>
<td>60</td>
</tr>
<tr>
<td>M</td>
<td>Statement of direction - canvas</td>
<td>86</td>
</tr>
<tr>
<td>N</td>
<td>Retrospective - canvas</td>
<td>88</td>
</tr>
<tr>
<td>O</td>
<td>Momento - card-set</td>
<td>90</td>
</tr>
</tbody>
</table>
Appendix A: Detailed overview of research methodology.

**Cycle 1: Exploratory research**

Following the problem definition the reduction of management attention was interpreted as framed as the problem of increasing autonomy of the team. Based on the initial analysis of the problem conducted by management of the department the problem was then split up into 4 themes (Table 1).

The dissection of the problem into these four themes formed the basis of interviews with team-members. Following a semi-structured format (appendix _) 6 interviews were conducted to explore the various facets of the problem from the perspective of the team. Added to this was data from informal interviews with members from around the department. In addition, 21 meetings were formally observed, combined with informal observations made while being embedded within the team. All data was collected anonymously, transcribed in keywords, coded in 58 codes and clustered in 12 emerging themes (Appendix C).

Parallel to the collection of data through interviews and observations a literature review on self-managing teams was conducted. In addition, a formal interview was held with an external expert on the subject. Combined with the findings from literature, observations and interviews a framework was created, presented in Theoretical framework, chapter 2.

**Cycle 2: Design research**

Following the first cycle a total of interventions were designed as part of the “Research through Design” methodology. These interventions were designed to resolve obvious roadblocks identified in the exploratory study, and were used to study the complexity of the context by manipulating the environment. As reasoned by Stappers and Giaccardi (2017) this increased understanding of the environment, what works and what doesn’t all contribute later on in the design of a solution to the challenge. Both the result of Research through Design (knowledge) and the process (design) deliver value simultaneously.

Part of this research cycles are the (un)intentional discussions that were prompted by the presence of the researcher. The results of these discussions are reported and contribute to the final design. The influence of the researcher on its environment is reflected upon in chapter.
Ownership.
The team should feel responsible for the development of the product, owning both successes and failures in order to solve them and implement solutions effectively (Wageman, 1997).

Scoping and internal alignment.
Deciding what to work and what not to work is a problem for the team, since implementation can be overwhelming. Making smart choices in order to deliver value is imperative for the success of both the product and department.

Autonomy.
If implemented successfully, an autonomous team offers a solution to reduce management attention and increase strategic flexibility and awareness by “open-sourcing” strategy to the team instead of management (Hackman, 1986; Langfred, 2007).

External alignment.
Communicating to external stakeholders is difficult given the number, their demands and the decisions that have to be communicated to them (Wageman, 1997).

Table 1  Overview of interview themes of exploratory study
Appendix B: Interview results

Internal alignment & scoping

The team is in large part aligned on their job: “the big assignment for us is to put Vanguard into production for the end user, to create a first minimum viable product that works and can be put into production with the functionality it has” (VGT3).

Every two weeks, what the team works on is determined collectively in “sprint planning sessions”. “We are perfectly capable to distribute the work amongst ourselves. (...) That’s a lot better, if you say you want to learn something, you can share this with the team and discuss to do some work related to that new skill” (VGT2). While this would be the ideal situation, the work is currently distributed more pragmatically; “we’re in a hurry, you know something about this so you work on this”.

All team-members mention a lack of a long-term vision on the use of optimizers. “Right now, I don’t see any vision, I have the impression we do this for one week and then another think the other week”. “We have really cool ideas about the future, but we don’t plan for long-term because every week something pops up and then we have to change a lot” (VGT4).

The team expresses their desire to have some more guidance in this aspect: “There is no plan for how Vanguard should be used by users” (VGT1). “It seems to me that decisions about this have been deliberately pushed forward by BCG” (VGT1). “I think it is odd that nobody from ODS gives alignment on the four optimizer projects” (VGT2). “I would like a more structured path, like a medium-term path where you can see, for up to like 3-6 months, in which direction are we going to productionalize.” (VGT4). This lack of vision further extends in about the mandate to make decisions about the vision. “There is a lack of focus, nobody makes these (architecture) decisions, like somebody should say; what do the users want and then we either we build a separate GUI or build an API and integrate it in other tools.” (OC1). “When you build a prototype you have to take into account how it will fit together (with other systems), this has not been done, and the problems are quite severe” (VGT5).

The author also noted during the interviews that there is a great deal of uncertainty about the development of a new GUI. In one of the user boards, Sander announced that he had hired a designer to redesign the user experience from scratch, but Moreover, the team experiences a lot of unclarity about the vision of the department, including the (long-term) involvement of BCG.

VGT2 remarks a lot of ad-hoc request come from outside of the sprint planning sessions, impacting the teams productivity; “recently we have a lot of meetings, so I don’t manage to put a lot of code into production” (VGT4). Some team-members remark these ad-hoc requests should be possible, but should be in discussion with the PO (VGT3) and in alignment with other PO’s (VGT2). The role of the PO in that regards should be to streamline stakeholder management (VGT1). “Ideally, the PO will shield you from direct nagging with stakeholders” (VGT2). Additionally, the team agrees that it’s the PO’s responsibility to set priorities; “We spend a lot of time on figuring out what we’re going to do and what the priorities are, what is important, this could probably be done by a product owner” (VGT4). “On the other hand, you want a balance (in direct contact
between stakeholders and team) otherwise the PO becomes the bottleneck” (VGT2).

“Now we also have a voice, this is important, we have some more influence” (VGT4).

For day-to-day work, the team agrees that a team-manager is not necessary (VGT2, VGT3) and that the team currently does perceive the PO to be a manager (VGT4). However, some team-members agree that a content-based team manager might become useful as the team starts to grow: “I think it would be good to have an expert manager, not a team-manager, but someone who, for example, leads all data scientists” (VGT3).

However, with such a manager it would be critical to leave the team autonomous. As the responsibility of this manager the management of resources (VGT2) and articulation of a vision (VGT4) are mentioned: “it seems to me that managing resources is one of the most difficult parts (for autonomous teams). Aligning your resources with (future) needs of the team to prevent the current situation (unbalanced team composition for the current project phase).” (VGT2). Another aspect of the manager would be to create space for personal development. “I personally haven’t done anything on personal development while it is very important” (VGT2). “What I miss in general is the feedback, I basically don’t get feedback; feedback on how I work” (VGT4).

On the day-to-day management one team-member remarks; “... maybe (we need a manager) through the beginning process, like managing to figure out how things should work” (VGT4). “It would be good to have a scrum-master, (...) but after a period with a scrummaster I think it would be really good to roulate who is the scrummaster for that sprint, I think that is also very educational.”

**External alignment**

Not everybody is enthusiastic about the way decisions with stakeholders are made. “Where decisions are being taken, that’s where there is a problem”. Across ODS decisions about the priority of different products are made in so-called “user-boards”. “If we would have been involved in certain decisions (about the planning of Vanguard), those decisions would have been made very differently. (During the user board) decisions have been made about things that (business owners) have no knowledge about. Promises have been made that are impossible to deliver.” (VGT1). VGT1 is also critical of the name; “We have the user boards, there is no user present but it is still called a user board. It’s a sort of steering committee meeting.”. In various discussions the user board is being referred to, but it appears that it is not always clear where decisions should be taken. Early on in the research the author noted that a discussion on the overarching vision of how ODS-products should relate stopped when it became unclear who would have the mandate to make those type of (architecture) decisions. During a similar meeting on the software architecture a team-member remarked: “we have to start at the user board, manage expectations and make (combining optimizers/architecture) a priority” (VGT2).

Yet another team-member noted about this: “It’s interesting to see how decision making differs between team and user board. (...) (Clients in) the user board have a bit more of a functionality focus, the team maybe focuses a little bit more on the technicalities of the product” (VGT6). Outside the user board the team is more enthusiastic. “We like to be open and transparent, (...) and it’s now becoming that ‘the business’ is also enthusiastic and actively thinking along, that’s very nice” (VGT3).

Team-members noted that BCG consultants have a way of framing their message in a way that sometimes hinders alignment with stakeholders later on. One example mentioned by all team-members is a PowerPoint presentation given in a user board
meeting that stated “Vanguard is ready to be used”. “No, this is absolutely not true. Vanguard is ready to be tested, there is a testable version of Vanguard. Maybe he means this, but by framing it this way you create a large problem in perception about the project status and direction” (VGT1). “This way you pain a nicer picture than reality, and with that create comments like “it is being used, why are you still working on this?”” (VGT2). “They set the perception by (not) including things in their diagrams, the size of block in a diagram also really matters. (...) In my view a lot of sugar-coating is being done in those presentations, to have as positive a message as possible.” (VGT1)

VGT2 expresses concerns that team-members spend a lot of time correcting the false expectations created in the user board. In correcting those expectations he remarks the team has to be pessimistic to weigh up against the optimistic message of BCG. “For stakeholders it is not that they are coding themselves, so it is a bit less tangible than seeing a new feature on the GUI. (...) In the beginning it was a little bit frustrating but now we have learned to explain it a little bit better. We have to explicitly if this is not going be fixed the whole optimizer doesn’t really makes sense; you have to really explicitly present it” (VGT4).

Ownership

None of the team-members consider what the current status of the project to be a Minimum Viable Product. “At this moment (Vanguards) is minimum viable, but it is not a product” (VGT1). One explanation for this that came up a lot during both formal interviews with team-members and informal interviews with ODS employees was confusion about the BCG terminology. “BCG terminology causes confusion” (VGT1). “For one reason or another BCG mixes up the order, they first make an MVP, then a prototype, this doesn’t make sense at all” (VGT2).

The team considers an MVP to be a strong foundation that can be built on (VGT1), tested (VGT3), is user friendly (VGT3, VGT4), and in an ideal situation is stable. It should have all functionality and offer value (VGT3) for the user (VGT2), that the MVP becomes large is not necessarily a problem (VGT1). “Vanguard is not minimum viable, what it needs to become viable is, in my opinion, the functionality of the engine, it should be able to solve more stuff” - VGT3. The team considers a prototype to come before an MVP (VGT1, VGT2, VGT3, VGT4, OC1), and considers the prototype to be a demonstration of a working principle (VGT2). “A prototype can be ductaped together, and after showing that it works you can start to build a real product. (...) A prototype is not used to gather feedback.” (VGT2).

The team agrees that there is a lot about Vanguard that needs to be fixed, noting issues with data processing (VGT4) and insufficient functionality of the current product (VGT3). VGT3 remarks: “I’m sure that you are not able to sell the current product to another airline, or put easily put it into production”. However, the team takes a constructive approach to making Vanguard work: “Large parts of the software need to be completely overhauled (...) But you shouldn’t throw away code for the sake of throwing it all away” (VGT1).

The team is generally not happy about the way the Vanguard project was transitioned from BCG to KLM. “Officially BCG involvement was decreased and KLM involvement was increased, but in practice it was ‘thrown over the fence’” (VGT1). “The entire transfer and attention from BCG for sustainable software, that is something we have to think about. BCG works by starting with a prototype, and when a user complains, they fix it, create a small experiment. At a certain moment in time it’s ductaped together, but it still
works, and then it is pushed to us. That’s how it went with Vanguard. (...) Development of robust software is not necessarily in the interest of BCG, they don’t experience the problems when it doesn’t work. I don’t think the people at BCG see this (robust software) as a problem, they don’t have to keep it running, maintain or build it out. (...)” (VGT2). This was echoed by an ODS Operations Consultant during an informal interview: “I experience no problems with the KLM team on architecture, but I foresee problems with the BCG guys. They don’t seem to concern themselves with architecture, while this is so important to do if you want to implement solutions and not only prototype.” (OC1).

“(Transitioning) could be so much easier if there was proper documentation and tests. (...) We need documentation on why certain decision are made, and we need to properly organise documentation, not in slides that are constantly changing but work with a single source of truth.” (VGT3). “The code needs to be cleaner and stable to do a handover, those types of things should be clear” (VGT4). “Maybe we should agree that we don’t do the handover before there are tests in there, before we know for sure that we not breaking things. So that we should have someone from KLM in the team for a couple months, together with the BCG team” (VGT3).

“BCG has built a proof of concept, and they don’t see it like that” (VGT1). “The way of working of BCG is to create a box, but from the beginning you have to pay attention to the content, this is not in the mindset of BCG” (VGT2). “I believe the contract between BCG and KLM is fixed fee, so it is in the interest of BCG to finish the job quickly.” (VGT1).

In terms of skills, all team-members talk about an imbalance of competences. “We are with 4 operations research experts, but to me it seems 3 (software) engineers and 1 operations research expert would be enough to complete the current phase of the project” (VGT1). As the reason for the low number of software engineers VGT1 remarks “there are no software engineers in the team because (MT) always hired data scientists”. As a reason for this the lack of domain-expertise of MT is pointed out: “why data science? In part because of incomprehension and ignorance” (VGT1). VGT1 further points out that in this regard “(MT) leans too much on the expertise of BCG (...) If they don’t make a distinction between operations research and data science, then MT doesn’t do that as well” (VGT1), “I think MT leans on BCG because of the incompetence in this field at KLM that used to be here. (...) Now this is different though, now there is super smart people walking around at KLM. (...) However, there is only so much smart people that you can listen to” (VGT1).

While the current assignment is to productionalize Vanguard, the team sees itself as the “optimizer”-team or “operations research-team” rather than the “Vanguard-team: “Sander said that we are not the Vanguard team, but the optimizer team” (VGT1). “We call it the Vanguard-team, but we are actually the Operations Research team of ODS” (VGT3). VGT1 remarks that although the team should be the optimizer team, his perception is that it is currently the “productionalizing” team. VGT3 adds to that: “I see us as responsible for how all optimizers are going to work. Part of this is the responsibility of BCG, but from KLM we are the ones that are closest to the joint venture. (...) We as Vanguard team have to take the lead in creating the architecture of all the optimizers (...) whether we will do that our self, or whether it would be better to hire someone to do that has yet to be determined, but I think we have the responsibility to initiate the discussion and ask a data architect to help us”.

“An MT-member said that a couple months ago he discovered there is a difference between data science and operations research consultant. That these are two different types of people. This is a thing, from our perspective this makes a big difference, also which of the two you are” (VGT1). Part of the reason that the team consists of data
scientists and not of software engineers is attributed to this apparent ignorance (VGT1, VGT3). VGT1 (data scientist) is more critical, remarking: “I did not need my studies to do my current day-to-day job (...) I joined Vanguard to do less data science and more Operations Research, I don’t do data science anymore, that’s an improvement, but there has been no Operations Research to replace it”. More team-members share this concern: “it would be nice to have more OR, more modelling and algorithmic thinking” (VGT4). VGT1 goes further by expressing his concern about a lack of intellectual challenge. “If you can’t do math, it’s a lot of fun to initiate interaction with your users. (...) But in strict agile methodology this job is mostly done by the PO, (...) so when this interaction goes away, I fear (the job) will start to become boring.” “In all honesty, currently people are not being used in that what they’re good at an in that what they most enjoy.”

Looking forward the team is less concerned; “I think MT is slowly starting to understand the difference between Operations Research and Data Science (...) What also helps is that now we will have a say in who will be hired, that those types of decisions are not made by a BCG project leader” (VGT3). “It’s a balancing act, you also expect an Operations Research to have software engineering skills, you need that (...) You expect to do a bit of everything, but if someone, for example, is really good at software engineering, maybe he then doesn’t have to be really good at consulting” (VGT3).

BCG is perceived to be doing the “fun work”. “BCG does the substantive part, that is then unfinished, and then we can do the software part” (VGT2). “Sentry, for example, is done by BCG guys, and they are working on the really cool models and people are kind of frustrated in the Vanguard team; why they do all the cool work and we have to fix their shit?” “We don’t do any math because that has already been done by BCG” (VGT1). This is not understood in the team; “I would like to build models, this is my background. (...) I don’t know why BCG builds models and not KLM. People from BCG are also talented, they have PhDs and things, but I think we’re not less capable of doing this kind of stuff. I think the product developed by us would be at least as good as theirs.” (VGT4). “We don’t even get the opportunity to show that we can (set up prediction models). To me that is very frustrating. That is why I am here, to do those types of things. And then we are asked to put things into production, for which we do not have the skills, and then you’re at a disadvantage because you’re doing something you’re not good at. (...) But, if apparently I’m not asked to do operations research, then it would not be a problem if I were to leave and was replaced by a software engineer” (VGT1).

VGT1 is also concerned about the optics of the difference between BCG and KLM. “An external consultant comes in and pushes something through the system in just 2 months, and then the KLM people have to take it into production, they are working on that for 8 months and seemingly nothing changes. Over time the MT will start to think; KLMers can’t do anything, those consultants can actually do stuff. While in 2 months something was hacked together with ductape, that is also something a KLMer can do, especially if you don’t have to pay attention to that at some time it has to be put into production”

Most team-members accept the situation as temporary. “Currently I get energy from software development, but I also already overheard other team-members say that they want to work on an OR problem” (VGT2). “If, one year from now I’m still not working on OR models, I don’t think I would enjoy it” (VGT3). But there is also reason for optimism; “if we put in a bit of effort into Vanguard, it starts to get really fun. Then, we can start to make real decisions. We’re going to run Vanguard in parallel 8 times with slightly different parameters, and see what comes out” (VGT3).

Not everything about BCG is negative. “A large part of our conversation is about how BCG delivers messy products, but they do it, and in record time. You would never have the
same thing if you had put the same number of KLM employees on the project.” (VGT2). “What BCG does, they do a lot aligning with KLM people in the beginning during the day, and when KLM people leave they start to code and actually put into production what they have understood. (...) It’s very efficient” (VGT4).

**Freedom**

The team talks about a strong team spirit (VGT2) and relying on each other: “I trust the team, knowing that they can do the job I can trust they’ll take the responsibility to deliver things. That’s how we should build on each other. (...) Within the team we understand each other’s’ strengths and weaknesses.” (VGT3).

In terms of (explicit) roles within the team some team-members are divided. VGT3, for example, is optimistic about the team and the progress that has been made, while VGT1 remarks to perceive the team to be in the “norming-phase”1 of teambuilding. The team largely agrees that a strict structure of roles like in the first year of the Vanguard project is not desirable; “a strict configuration like before wouldn’t help (...) but it would be good to discuss team-member responsibility explicitly” (VGT1). “In silos best practices (about coding) are not shared. You want to have cross-fertilization in the code, so you have to work on everything together” (VGT2). “Currently, team-member 1 knows a little bit more about A, while team-member 2 knows a little bit more about B, but I wouldn’t call it roles” (VGT3).

The team agrees that autonomous teams are a good thing, provided that the right skills and competences are in place (VGT2, VGT1). VGT3 is enthusiastic and remarks “it starts to move”. “(having to do everything yourself) triggers you to work harder in a way, to think in a higher perspective than just being a developer.” (VGT4). “Nothing is standing in the way of becoming an autonomous team, only if there was a manager, but right now there is nothing. I think we manage to prove (to the business) that we’re trustworthy, we’re managing to get into this workflow” (VGT4). “After a while I think we can become much more productive and faster in making decisions and going in a direction.” (VGT4)

With one exception, the team agrees Sander is their boss. The team experiences a lot of trust from MT (VGT3, VGT4) and appreciates the Sanders’ leadership style; “I like how (Sander) approaches us, he says you can do whatever, as long as you think it is the best option” (VGT4). “I think that if we would mess it up, Sander would have our backs. (...) Through coaching, he wants to push the team forward, instead of correcting us, that builds trust.” (VGT3).

However, what the team misses is structure in the alignment with Sander and clarity about authority; “currently, there is no official alignment with Sander , I think some people just go to him and ask questions, sometimes he comes to our table but it’s not an official thing (...). It would be nice to have at least an alignment with Sander where you can align on the mid-term goals, or give some feedback” (VGT4). “I think it is very important what Sander and Maarten think, to know what they want and where they want to go. (...) Not in steering what the product should be, but in terms of understanding how Sander sees Vanguard (...) If Sander pictures Vanguard to be a certain way, that’s important to know, then you can say ‘but this is not what we’re doing right now, if we continue this way we will end up here while you expect us to end up over here’” (VGT2).

Regarding ad-hoc requests from the business, three team-members place question marks about the relationship between their PO and who they perceive to be his boss, citing a hypothetical situation in which his boss overrules a priority set by the PO in favour of
the request. Furthermore, ad-hoc requests like indicate to the team that the business is not aligned on priorities. “If there is a discussion, it should be clear who makes the final decision” (VGT2).

Similarly the team indicates a lack of clarity on the vision for ODS (VGT2) and involvement of BCG (VGT4, VGT2). “BCG does stuff, but how that is being coordinated exactly is unclear to me” (VGT2). “Shouldn’t we be involved earlier in those (optimizer projects)? Shouldn’t we know a little bit more about BCG?” (VGT2).

As to the presence of BCG the team notes several reasons why they think BCG is hired. “I know why they are here; to quickly build a prototype. For BCG to sell it to others and have a nice showcase. For KLM because they can gather speed, have instant good resources that work hard, temporarily for a reduced fee, or even free. (...) Besides that, everything an expensive consultant says will be believed. (...) Because they talk on various levels of the organisation, you can use them as a communication channel. It’s weird but effective.” (VGT2). “BCG creates prototypes, that’s what I understood, and also business transformations. They bring together different departments and then you come together with an optimizer that has an influence on all departments. Other than that, why should they be here?” (VGT4). This unclarity about ODS and BCG has been often been noted by the author in informal conversations with colleagues throughout the department.

“I think it’s a really good team, the people like each other but everybody also has their own personality and vision, but it’s not mutually exclusive.” (VGT4) “I notice that as a team we are getting more used to each other, to me that seems like quite a feed since we were simultaneously cleaning up a mess” (VGT3). “I think as a team we learn a lot, the hectics also teaches you to stay calm” (VGT4)
<table>
<thead>
<tr>
<th>Category</th>
<th>Code</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Alignment (external)</strong></td>
<td>Meetings with stakeholders</td>
<td>6</td>
</tr>
<tr>
<td><strong>Alignment (internal)</strong></td>
<td>Distributing work amongst team</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Space for personal development</td>
<td>7</td>
</tr>
<tr>
<td><strong>Autonomy</strong></td>
<td>Autonomous teams</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Conditions for autonomy</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Freedom</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Gains from autonomous teams</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Impediments for autonomy</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Managing resources; autonomy impediment</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Structure</td>
<td>10</td>
</tr>
<tr>
<td><strong>BCG / KLM</strong></td>
<td>BCG vs KLM work assignment</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>BCG way of working</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Correcting expectation created by BCG</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Culture differences</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>Expertise BCG</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Framing by BCG</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>KLM/BCG transition</td>
<td>16</td>
</tr>
<tr>
<td><strong>Definition</strong></td>
<td>MVP - definition</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Prototype - definition</td>
<td>7</td>
</tr>
<tr>
<td><strong>Future way of working</strong></td>
<td>Desired way of working</td>
<td>5</td>
</tr>
<tr>
<td><strong>Leadership</strong></td>
<td>Lack of clarity on BCG involvement</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Lack of vision on ODS</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Lack of vision on VG/optimizers</td>
<td>37</td>
</tr>
<tr>
<td><strong>Management</strong></td>
<td>Alignment with management</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Lack of expert knowledge from MT</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Leadership paradox</td>
<td>1</td>
</tr>
<tr>
<td>Category</td>
<td>Issue</td>
<td>Count</td>
</tr>
<tr>
<td>---------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>Personal frustrations</td>
<td>Feeling of not being heard</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Job description mismatch</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Lack of opportunity</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Lack of recognition</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Motivation</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Need for intellectual challenge</td>
<td>3</td>
</tr>
<tr>
<td>Project - general</td>
<td>Clarity team responsibilities</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Code quality</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Current project focus</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Project assignment</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Project history</td>
<td>3</td>
</tr>
<tr>
<td>Scoping</td>
<td>Ad-hoc requests</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Decision making</td>
<td>13</td>
</tr>
<tr>
<td>Team dynamics</td>
<td>Learning as a team</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Personality traits</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Productivity</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Required job profile</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Responsibility that comes with autonomy</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Role of PO</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Team - norming phase</td>
<td>2</td>
</tr>
<tr>
<td>Team composition</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Team processes</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Team roles</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Team spirit &amp; trust</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>
### Appendix D: Overview of meetings

<table>
<thead>
<tr>
<th>Datum</th>
<th>Meeting</th>
<th>Type</th>
<th>Date</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/7/2018</td>
<td>Kick-off with team</td>
<td>Team-meeting</td>
<td>6/20/2018</td>
<td>Interview 9</td>
</tr>
<tr>
<td>5/8/2018</td>
<td>Standup</td>
<td>Team-meeting</td>
<td>6/28/2018</td>
<td>Urgent sprint planing VG</td>
</tr>
<tr>
<td>5/8/2018</td>
<td>Architecture alignment</td>
<td>Team-meeting</td>
<td>6/28/2018</td>
<td>Scrum prep meeting</td>
</tr>
<tr>
<td>5/8/2018</td>
<td>Sprint-session team</td>
<td>Team-meeting</td>
<td>6/29/2018</td>
<td>Interview 10 (MT)</td>
</tr>
<tr>
<td>5/9/2018</td>
<td>Alignment F720</td>
<td>ODS meeting</td>
<td>6/29/2018</td>
<td>Interview 11 (MT)</td>
</tr>
<tr>
<td>5/9/2018</td>
<td>User board</td>
<td>Workshop</td>
<td>6/29/2018</td>
<td>MT evaluatie VG team</td>
</tr>
<tr>
<td>5/16/2018</td>
<td>Stijn / Sander alignment</td>
<td>Coaching</td>
<td>7/3/2018</td>
<td>Sprint demo</td>
</tr>
<tr>
<td>5/16/2018</td>
<td>PO alignment</td>
<td>ODS meeting</td>
<td>7/6/2018</td>
<td>Sprint retro + planning</td>
</tr>
<tr>
<td>5/18/2018</td>
<td>Wizard of Ops (@Cityhopper)</td>
<td>Team-meeting</td>
<td>7/16/2018</td>
<td>Sprint review</td>
</tr>
<tr>
<td>5/23/2018</td>
<td>Interview 1</td>
<td>Interview</td>
<td>7/19/2018</td>
<td>Stijn / Sander</td>
</tr>
<tr>
<td>5/23/2018</td>
<td>Interview 2</td>
<td>Interview</td>
<td>7/20/2018</td>
<td>Stijn / Ocky</td>
</tr>
<tr>
<td>5/30/2018</td>
<td>Interview expert</td>
<td>Interview</td>
<td>7/23/2018</td>
<td>Sprint review</td>
</tr>
<tr>
<td>6/7/2018</td>
<td>Interview 3</td>
<td>Interview</td>
<td>8/27/2018</td>
<td>Sprint review</td>
</tr>
<tr>
<td>6/7/2018</td>
<td>Interview 4</td>
<td>Interview</td>
<td>8/27/2018</td>
<td>Sprint retro + planning</td>
</tr>
<tr>
<td>6/7/2018</td>
<td>Interview 5</td>
<td>Interview</td>
<td>8/28/2018</td>
<td>Interview 11 (MT)</td>
</tr>
<tr>
<td>6/7/2018</td>
<td>Architecture alignment 2</td>
<td>Team-meeting</td>
<td>8/31/2018</td>
<td>MT evaluatie VG team</td>
</tr>
<tr>
<td>6/8/2018</td>
<td>Stijn / Ocky</td>
<td>Coaching</td>
<td>9/10/2018</td>
<td>Sprint demo</td>
</tr>
<tr>
<td>6/12/2018</td>
<td>Restaurant 1</td>
<td>Interview</td>
<td>9/10/2018</td>
<td>Sprint retro + planning</td>
</tr>
<tr>
<td>6/14/2018</td>
<td>Interview 12</td>
<td>Interview</td>
<td>9/11/2018</td>
<td>OpX workshop</td>
</tr>
<tr>
<td>6/19/2018</td>
<td>Interview 7</td>
<td>Interview</td>
<td></td>
<td>(45 x) daily stand-up</td>
</tr>
<tr>
<td>6/19/2018</td>
<td>Interview 8</td>
<td>Interview</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/19/2018</td>
<td>PI event VG</td>
<td>ODS meeting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/20/2018</td>
<td>MT overleg 1</td>
<td>ODS meeting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/20/2018</td>
<td>Stijn / Barend</td>
<td>Coaching</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/20/2018</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/23/2018</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/27/2018</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/29/2018</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Daily Stand-up**

<table>
<thead>
<tr>
<th>Datum</th>
<th>Meeting</th>
<th>Type</th>
<th>Date</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/7/2018</td>
<td>Stijn / Sander</td>
<td>Coaching</td>
<td>6/20/2018</td>
<td></td>
</tr>
<tr>
<td>6/20/2018</td>
<td>MT overleg 1</td>
<td>Coaching</td>
<td>6/20/2018</td>
<td></td>
</tr>
</tbody>
</table>

---

**Informal Meetings**

<table>
<thead>
<tr>
<th>Datum</th>
<th>Meeting</th>
<th>Type</th>
<th>Date</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/30/2018</td>
<td>Interview 8</td>
<td>Interview</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/13/2018</td>
<td>Restaurant 1</td>
<td>Interview</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/14/2018</td>
<td>Interview 12</td>
<td>Interview</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/19/2018</td>
<td>Interview 7</td>
<td>Interview</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Informal Meetings**

<table>
<thead>
<tr>
<th>Datum</th>
<th>Meeting</th>
<th>Type</th>
<th>Date</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/20/2018</td>
<td>MT overleg 1</td>
<td>ODS meeting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/20/2018</td>
<td>MT overleg 1</td>
<td>Coaching</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Appendix E: Interventions overview

<table>
<thead>
<tr>
<th>Intervention</th>
<th>Team Identity</th>
<th>Organisation</th>
<th>Culture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Designed to resolve/contribute to</td>
<td>Owner</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Team</td>
<td></td>
<td>Stijn, VG</td>
<td>Ongoing</td>
</tr>
<tr>
<td>Team</td>
<td></td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Team</td>
<td></td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Identity</td>
<td></td>
<td>VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Discussion, process and terminology</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discussion: team identity</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discussion: sales</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Intervention: Scrum implementation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Intervention: Product vision statement</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Result: Product roadmap</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>Status</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Management</td>
<td>In progress</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VG/Management</td>
<td>Ongoing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stijn, Management</td>
<td>Completed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Management</td>
<td>Ongoing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stijn, Management</td>
<td>Completed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Management</td>
<td>In progress</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stijn, Management</td>
<td>In progress</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stijn, Management</td>
<td>In progress</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Organisation</th>
<th>Designed to resolve/contribute to</th>
<th>Owner</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Ongoing</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn, VG</td>
<td>Completed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Culture</th>
<th>Designed to resolve/contribute to</th>
<th>Owner</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Ongoing</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Design</td>
<td>Framework</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Vision</td>
<td>Strategy</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
<tr>
<td>Trust</td>
<td>Design</td>
<td>Stijn</td>
<td>Completed</td>
</tr>
</tbody>
</table>
Appendix F: Interventions product team

Intervention: statement of direction.

The implicit goal poses a challenge for collaboration. To collaborate successfully, the objective must be clear. From discussions it can be inferred that, at least concerning the product they develop, the team-members have more or less the same vision. Making it explicit, similarly to the use of a ‘boundary object’ in design (Stevens, 2013), gives the vision ‘form’; it facilitates discussion as all team-members are able to point at the thing they are talking about. Making the vision explicit can be a tool to hold yourself accountable (reference). The assignment is to productionalize the product, the goal is challenging, a team-task and internalizable. It’s also time-bound, although this does not become immediately apparent from interviews. The thing about that the product teams’ goals is missing its clarity. Wageman (1997) introduced a “statement of direction” for this, stating that a goal should be 1) clear and simple and 2) specific in ends but not in means. She found that a crucial error in self-managing teams is failing to set direction. In an attempt to discuss the objectives of the team and create a statement of direction the researcher initially planned an “off-site” workshop with the team to define and visualise this vision, as well as discuss team responsibilities. The off-site never happened, but there came an opportunity to make the vision of the product more explicit.

Done. One day during the study management pressed the team to deliver a (highly scoped) working product within two weeks. During an emergency sprint planning session the researcher and team defined for that coming month exactly what the delivered functionality per product module was going to be. The researchers captured this for each product module in a single sentence. Each ‘version’ of the module was visualised in a roadmap. All visuals were combined into a PowerPoint-presentation and send to all stakeholders involved.

Result. The sentence was constructed in a meeting with the team facilitated by the researcher. Three versions were created for the three modules of the product (Figure 8). The researcher summarized these versions into a PowerPoint presentation, together with 1 team-member. The PowerPoint was sent to the management team, who forwarded it to stakeholders involved. Six weeks after the initial meeting the team reflected on the version sentence as useful. However, not all the elements of the version were in delivered by the team.

Findings. Although considered useful as an idea, to be effective the version statement must play a role in the scrum process; for example, it could be visible as a constant reminder. Furthermore, team-members should feel responsible for delivering the version by a set deadline. Creating ownership could be done by drafting the statement together in a meeting.
**Intervention: Scrum.**

To speed up the adoption of team norms the researcher stimulated the team to adhere more strictly to a standard way of working. Arguing that an agreed on way of working reduces ambiguity around roles and responsibilities and timing of conversations could benefit the focus of the team, and more in the short term the reduction of ad-hoc requests by explicitly defining roles. As a way of working “Scrum” was agreed on by the team. Scrum is a method for software development that follows the agile development philosophy. The method is also used by other teams in the department and the airline.

Done. Together with the team a plan of action was created. This involved the researcher adopting a (temporary) role of scrum master, the organisation of a scrum-workshop by an expert, and the search of a permanent experienced scrum master. For a period of two weeks the researcher took on the role of scrum master, followed by a workshop organised by an experienced scrum master. During this workshop a temporary scrum-master was introduced to the team, who facilitated subsequent meetings. Later a permanent scrum master was appointed to the team.

Result. 8 weeks after starting with the introduction of the role of scrum-master the team has grown into the meeting cycle of scrum. The team seems a lot more focused in development (note). Every two weeks the team demonstrates the product to a group of users and stakeholders. Parallel to the adoption of scrum the team has changed development strategy to develop alongside the users.

Findings. From the experience of the researcher it followed that being a scrum master is difficult and requires some experience (note), particularly when the team is first starting to use scrum. Although each meeting has a specific purpose, it is easy for a discussion to ‘spiral out of control’ beyond the scope of that meeting. Particularly during daily stand-ups (short 15 minute meetings to update team-members on individual progress) this is likely to occur. This indicates that everyone, not only the scrum-master should be aware of the purpose of the meeting and feel comfortable stepping in when this lost of out sight. The structure of the scrum process seems to greatly improve the focus of the team. Ad-hoc meetings have drastically reduced. This could be partly attributed to the roles and responsibilities of different team-members being discussed in the meetings of scrum.
Appendix G: Interventions organisational context

Given the complexity of the larger scale of the department and limited time available for the study interventions at the level of the department were more difficult and ‘slow’ to implement. The objectives of these interventions were to resolve immediate roadblocks (vision of product system) and to challenge and facilitate management in creating organisational clarity. Next to this the researcher experimented with different interventions for both the team and the department, to reduce feelings of “them-and-us”, following the blueprint of a generative culture. With interventions and discussions the researcher challenged team members and members of the organisation to start a conversation with each other about culture. An intervention was done where all the members of the organisation were given a workshop in sketch-noting in an attempt to the first outline of a common language. A language that supports associative thinking and in that way entrepreneurial behaviour. Team-members were challenged to be proactive.

Findings. The use of a vision that is pragmatic can be used as a boundary object to align with each other, not just within a team, also between teams (note). Secondly, this example, although just one instance, seems to indicate that when prompted to be proactive there is a willingness to take the lead in solving problems.

Intervention: product system vision.

Done. From research it followed that an unclear vision on the system of products complicated product development for the team. To resolve this roadblock the researcher challenged the team to take the lead in the development of this vision.

Result. An internal consultant of another team was appointed project lead and together with the researcher created a first vision for the system of products (note). This vision was presented by the consultant and subsequently led to weekly alignment meetings between teams involved, as well as a supervisor (consultant) to ensure the quality of the products and interactions with each other (note).
Intervention: sketch workshop.

A key aspect of design is the creation and use of boundary objects (Stevens, 2013). Taking the form of products, prototypes, presentations, a boundary object is in effect anything that can be pointed towards (Stevens, 2013). As such it facilitates a conversation; aligning participants with a concrete reference point for the words and images they project onto it; it makes conversations explicit. One type of boundary object used in this study is the use of visualisations to learn about the problem (Gudiksen, 2012). Used by the researcher as a tool to structure thinking, the idea sprung up amongst a group of members of the department and the researcher to teach the basics of this technique to all the members of the organisation as the start of a ‘common language’, with the idea of opening up a conversation about strategy.

Done. A workshop was designed that outlined the basic techniques of sketch-noting. One 2-hour was given to a group of 15 participants, all members of the department received a smaller 20-minute workshop. The workshop explained the use of drawings, taking participants by the hand to learn mind-mapping, drawing icons, visualising situation using metaphors and drawing people. An overview of the workshop slides can be found in appendix _.

Results. Participants were enthusiastic, asking for more workshops and more elaborate explanation of specific techniques. A channel was created on the messaging-application used by all members to encourage each other to draw. Over the following period drawings popped up in the office and the researcher was invited to support others in visualising their projects.

Findings. As a ‘common language’ sketch-noting seems interesting as is picked up by individual members of the organisation, but not by everybody. It seems however that the language is understood and accepted by everyone; this follows from drawings being used in presentations to explain things. The workshop itself was found to be a lot of fun and could in itself provide a lot of value for group development of the department.
Intervention: townhall meeting.

Research showed that members of the organisation were not clear on the vision and structure of the department. The researcher designed a ‘platform’ to stimulate conversations about culture and organisation, in the form of a “townhall-meeting”. A meeting where management is invited onto a stage to answer questions from all the members of the organisation. The idea of this townhall meeting is to provide a platform where everybody comes together, for management to share updates that concern everybody, and for the members to ask questions that concern everybody. As such it is an instrument to facilitate communication to create openness and build trust (Hakanen & Soudunsaari, 2012).

Done. The idea was proposed to management and accepted, a first meeting was organized as part of a training the department was scheduled to receive. The first half of this training a professional facilitator invited management onto the stage to discuss with the department three examples of behaviour of teams in and around the department (Appendix _). Members were then encouraged to sit at someone else’s desk and reflect on whatever they wanted to reflect on, guided by a canvas designed to structure the reflection (Appendix _). In another exercise team members would throw a string of wool to connect over similar feelings that each member would shout out upon receiving the string. This created a network of connected people. The exercise was used by the facilitator as a wake-up call and encouragement to start to form a team.

Result. The meeting was confrontational yet productive. A discussion about culture across the department was triggered. 34 reflection forms were collected, the results of which are part of the research section of this report (appendix _). The training was concluded with the agreement to work together to become a team, with members of the organisation signing up to organise drinks, townhall session, informal lunches to connect.

Findings. Received mostly positively (note), the townhall could be a powerful instrument for cultural change. Important, however, is to schedule it regularly in order for it to become effective. Some members of the organisation were sceptical about this. As a one-time meeting it triggered a discussion that in itself could be valuable.
Intervention: challenge the team.

Done. As part of creating a pro-active mindset the team was challenged to take the lead in resolving roadblocks, instead of them being resolved. This way interventions to create vision on the product and vision and meeting structure to discuss the system of products was initiated. The team was challenged to discuss process terminology used by external consultants not only amongst airline employees, but together with external consultants.

Result. The terminology was discussed with external consultants and management and eventually changed (note). Prompting the team to create a vision resulted in a vision and meeting structure to periodically align between teams on the technicalities of the system of products (note).

Findings. Prompted to take the lead it seems there is certainly willingness, even enthusiasm to tackle obstacles.
Observation: demo.

Not triggered by the researcher, management started organizing bi-weekly demo sessions where two teams are invited to present their project and progress. The result is a bi-weekly event where all members of the department gather, both airline and external consultants, to recognize the project of another team. It is an opportunity to get to know each other professionally, it is an opportunity to collectively admire the progress of the organisation and an opportunity to celebrate the expertise of colleagues.
Appendix H: Discussions

Various discussions were (un)intentionally triggered upon sharing the findings of this research with management and the product team.

External consultancy culture integration. Discussions between the researcher and management resulted in a clear mandate from the external consultancy management for its consultants to integrate culture with that of the department. During the town-hall meeting airline consultants were encouraged to do the same.

Discussions on organisational culture. The presence of the researcher triggered many informal conversations with members of the organisation, discussing ideal culture. During this discussion the researcher to a deliberate approach to challenge participants not to complain, but to be constructive in their discussions; challenging them to be pro-active upon identifying obstacles or adopting a best practice. One of the ideas encouraged was to create a visual overview of “weird” skills of the members organisation (skills like “formula 1 expert”, “salsa dancer”, etc.) and to organise a pub-quiz evening around the themes of those skills. These initiatives are ongoing (note). Organisational culture is also an important item on the agenda of the management team (note).

Discussion: MT role implicitness. The finding that there is a potentially problematic disconnect between what management says (self-managing teams) and what it does (urgent request) was shared with management. This sparked a discussion about the feasibility of self-managing teams and organisational structure of the department. It also led to a change in behaviour from management where a second time an urgent request was send to the team this was done by sharing the concern, instead of a demand.

Discussion: organisational structure. The discussion on the behaviour of management sparked a discussion on the organisational structure, something discussed at length during an off-site meeting of the management team. Although this meeting ended without a clear conclusion (note), the discussion was later followed up by one of the managers with a temporary organisational chart (note).

Discussion: manifest ODS. The presentation of the findings of the preliminary research triggered a discussion about the need of a vision, and the level of explicitness. An external communications consultant was hired to help draft a manifest of the department, to be shared with both employees and clients. The crafting of this manifest is part of the final design of this project, further described in "3.2 Design: ODS" on page 66.

Observation: on-boarding. Following a series of new hires management concluded that the on-boarding process of the department was unclear. Management initiated a project to collect all the information required for the on-boarding of a new team-member into one place. This is not only useful for new hires, it can also be used as a reference point for existing employees. This project is ongoing.

Idea: visual overview of ODS portfolio. To create clarity about the (overlapping) scope of the various projects of the organisation the researcher initiated a project to visualise the different projects of the department into 1 drawing. Starting with visualising two key technical systems of the department, the first version of the visualisation included only two projects. The project is ongoing. The idea is to print the visualisation and have other teams draw their projects in relation to the depicted projects.
Appendix I: Expert interview transcript

Interviewer: je bent bekend met ODS, in december hebben we een leuk gesprek gehad waarbij je zei; je moet het gewoon gaan doen (een team zelfstandig maken). Dit was voor mij de trigger jou te vragen. Kijkend naar ODS, wat zie jij als de grootste obstakels voor een team als bijv Vanguard om echt autonoom te worden?

EX1: dan eerst vragen wat is autonoom team.

Interviewer: is dat wenselijk, wanneer wel wanneer niet

EX1: er zijn twee absolute voorwaarden wil je autonomie in een organisatie inbrengen, in een teamorganisatie inbrengen. En dat is 1 het principe dat je niet vanuit anarchie werkt, anarchie is complete regelloosheid werkt en volledige inrichting volgens je eigen ideeën en richtingen. En autonomie is hier in organisatiedenken altijd gekoppeld aan een kader.

Interviewer: in de vorm van..

EX1: altijd ingekaderd in de vorm van een purpose of een visie document of een strategisch kader, autonomie in organisaties is nooit de vrijheid die het woord. Zogenaamd in zich draagt. Autonomie in organisatie is een soort contradictie.

Interviewer: is dat een probleem?

EX1: dat hoeft niet, als je maar, ook weer framing, zorgt dat de juiste definities worde gehanteerd en de juiste verwachtingen worden neergezet en als je praat over twee voorwaarden als je praat over het autonoom krijgen van groepen mensen of sterker in performance en versterken van eigen verantwoordelijkheid en ondernemende gedrag wat daarbij hoort. A het kader nodig, een kader waarbinnen die vrijheid kan worden ingevuld, dat heeft organisational clarity. En een tweede element is op z’n minst een aantal afspraken hebt over de inrichting van die organisatie, we noemen dat technische competenties. De verantwoordelijkheden, taken, bevoegdheden, processen, systemen. Die twee zijn niet teovallig ook die twee die te maken met de randvoorwaarden rondom leadership intent. Leadership intent is een besturingsfilosofie die eigenlijk uitgaat van het versterken van de autonomie van individuen en teams. Het autonome handelen. En de essentie erachter is dat, leadership intent is een besturingsfilosofie, geen management stijl, subtiel verschil. Duis je kan zeggen van KLM is nu gekozen om als management stijl, leiderschap stijl coachend leiderschap te implementeren, dat is een leiderschap stijl. Daar zit 1 niveau boven namelijk de besturingsfilosofie. Namelijk hoe wil je dat leiders zich gedragen in relatie tot het realiseren van doelstellingen en bijdrage àan de purpose. Dan kun je allerlei technische of vaardigheidselementen toe gaan voegen namelijk coachend leiderschap etc. Je hebt nodig daarbij de besturingsfilosofie, de besturingsfilosofie zou moeten zijn dat je als leider je mensen in staat stelt om autonoom zelfstandig te kunnen handelen en groeien.

Interviewer: naar jouw mening, is dat op dit moment het geval?

(...) Coachend leiderschap bijv, coachend leiderschap is niets anders dan het begeleiden van een proces terwijl bij besturingsfilosofie zegt ik wil dat je op een bepaalde manier gaat werken.

Interviewer: en op welk niveau moet daar invulling aan gegeven worden? Als we het over
KLM hebben, op welk niveau

EX1: dat kan elke leider voor zich doen, alleen in de hierarchie van de KLM loop je elk moment als leider van 1 hierarchisch moment aan tegen het leiderschapsidee van de volgende, dus in de meest ideale situatie.

Interviewer: bij de CEO beginnen

EX1: ja maar dat geldt voor elk concept wat je doet. Het leiderschaps model, of de besturingsfilosofie zou een besturingsfilosofie moten zijn die generiek is, die hoort bij ons. Als wij pioneers by heart zijn, als wij de customer experience hoog achten, als we onze mensen willen ontwikkelen en empoweren dan moet je veel verder gaan dan alleen maar coachen.

Interviewer: ik kan niet KLM gaan verbouwen, laten we een klein beetje terug zoomen naar ODS. Hoe is dat hier qua besturingsfilosofie en misschien ook managementstijl binnen ODS? Is dat al een stap in de juiste richting?

(...)

Interviewer: is dat dan, de teams hier zijn nog niet autonoom, hoe zou jij dat zien? Dat is mijn observatie, waar ik naartoe wil; hoezo niet, hoezo zijn ze niet autonoom? Tweede vraag die daarachter schuilt is het überhaupt wenselijk dat teams autonoom zijn?

EX1: binnen organisational clarity, dit is waar we voor gaan, dit is onze purpose, dit is waar we voor gaan, dit zijn onze strategieën, en binnen die strategieën zou je de autonomie kunnen inrichten. Daar zit ook meteen de contradictie, heet het wel een autonoom niet, maar dan handelt het autonome team binnen een besturingsfilosofie dat er vanuitgaat van een bepaald kader waarbinnen je die vrijheid van handelen hebt. Dat noem ik het improvisatiespectrum. Het improvisatiespectrum is dat deel waarin je niet regesseerd en afgesproken vrijheid van handelen hebt op basis waarvan je op eigen initiatief richting mag geven, maar wel, en daarom heet het improvisatiespectrum, het is niet free for all. Het is niet zo dat,

Interviewer: nee er is een bepaald kader. Hoe zou zo’n kader eruit moeten zien?

EX1: jullie gebruiken, de KLM gebruikt de OGSM. De OGSM is een kader waarbinnen je de besturingsfilosofie verder kan inrichten.

Interviewer: is die voldoende effectief? In jouw ogen

EX1: ja, in mijn volledige overtuiging wel. Niet als je hem als obligate oefening invult en onder in de la legt daarna, maar wel als je hem in je cyclus van planning en control meeneemt en wholeheartedly met je besturingsverantwoordelijke hebt ingevuld.

Interviewer: dan kan hij gidsen, dan kan hij sturen.

EX1: het gaat om push authority where the information is. Dat is een uitspraak van iemand die leadership intent als besturingsfilosofie heeft ingebracht. Dat betekent, als je nou uit gaat van het feit dat als mensen informatie hebben dat ze ook de autoriteit moeten hebben om daarmee om te gaan.

Interviewer: dus om een expertise echt tot uitvoering te kunnen brengen?
EX1: als je een samenballing van informatie hebt, het meest simpele voorbeeld is een nachtportier in een hotel die te maken krijgt met een gast die ontevreden is, simpel voorbeeld. Die op dat moment alle informatie heeft van het hotel en de gast, die zou de authoriteit moeten hebben om op dat moment te zeggen "meneer u krijgt morgenochtend van mij gratis ontbijt aangeboden, zo vervelend". Nu moet hij naar zijn manager die er niet is en dan een papiertje invullen, push authority where the information is. En niet "push information where the authority is" wat eigenlijk in een traditioneel bedrijf heel vaak gebeurt. Want daar gaat informatie naar een bepaald authoriteitsniveau en daar wordt een besluit genomen.

Interviewer: mooie, gelijk twee vragen. De ene: hoe zorg je, dat je authoriteit hebt betekent dat je beslissingen kunt nemen, beslissing nemen heb je informatie voor nodig, informatie is niet alleen informatie die jij op dat moment zichtbaar hebt, maar je moet ook informatie hebben die eventueel andere mensen die impact gaan hebben van jouw beslissingen.

EX1: zou kunnen

Interviewer: Hoe zou dat eruit moeten zien?

EX1: hoe dat eruit moet zien. Het is nog een tweede vraag, het feit dat het toegankelijk is en beschikbaar is, is de primaire vraag, als je maar zorgt dat je de informatie die nodig is om tot een besluit te komen voorhanden hebt. Als een passage employee bij een instapproces onvoldoende informatie heeft om te besluiten een passagier alsnog door te laten of te wachten, dan kun je ook niet de authority daarnaartoe brengen. Als je de authority brengt naar de passage employee moet je ook zorgen dat de passage employee al die informatie heeft of de toegankelijkheid tot die informatie heeft om die besluiten te kunnen nemen.

Interviewer: En dat beantwoord eigenlijk al mijn tweede vraag. Want de tweede vraag zou zijn als je een project hebt of iets op gaat zetten heb je verschillende fases in ontwikkeling, waarin de fases verschillende expertises, verschillende rollen de hoofdmoon hebben. Op het moment dat je een idee hebt zijn dat misschien creatieve mensen, die denken hoe zou dat idee eruit moeten zien. Daarna zijn de data scientists, die denken hoe zou het model eruit moeten zien, en dan de software ontwikkelaars en dan de business, mensen die de software moeten gebruiken. dus hoe zorg je ervoor dat de autoriteit doorstroomt door die verschillende fases heen? Dat daar geen conflict zit. Dat een data scientist geen beslissing neemt die later negatieve impact heeft op een developer, op het werk dat een developer moet gaan doen.

EX1: dat gaat ook weer over technical competences die op orde moeten zijn. Je moet dus ook weten hoe de verhoudingen in het proces in elkaar zitten, je moet weten wat de verbindingen zijn. Je moet de logica snappen van de procesvoering, de impact van besluiten, dat is allemaal technical competences. Dat zeg ik wel zo snel, maar dat is een ontwikkelproces.

Interviewer: het is complex

EX1: het overdragen van besturingsvrijheid ofwel autonomie, het overdragen van besturingsvrijheid vraagt om een drietal dimensies te ontwikkelen. Dat is de dimensie van ‘samengwerking en communicatie’, dat is de dimensie van ‘strategie en helderheid’ en dat is de dimensie van ‘technische competenties’. De eerste gaat over van hoe gaan we eigenlijk met elkaar om en wat zijn parten in onze communicatie en wat doen verschillen in persoonlijkheden en wat doet teamdynamiek. Tweede heeft te maken
met OGSM perspectieven, het kader, de strategieën, de visie, de missie. De derde heeft te maken begrijpen we wel binnen welke procesvoering we zitten, hebben we daar goed overleg over? Kunnen we dat in zijn vertakkingen begrijpen, weten we de afhankelijkheden, dat is een lerend proces, dat is niet af, dat is eigenlijk nooit af. Zoals geen van deze drie elementen af zijn. De communicatiestijlen en dynamieken kant is nooit af want er is altijd een veranderende teamsamenstelling, altijd. De visiestructuur is nooit af want er zijn altijd externe omstandigheden of omstandigheden intern die de prioriteitsverschuiving met zich meebrengen of die na een periode […]. Dus het is nooit af. Dat maakt het ook interessant natuurlijk. Het is net zoals leven. Leven doe je. Daar kun je allerlei regels omheen denken, je doet dingen in het moment. Met agile net zo, je kijk je weer terug en kijk je vooruit en heb je weer een nieuw moment. En wat je doet met autonomie versterken en met leadership intent ook is dan versterk je de kracht van mensen in het moment, los het maar op, want binnen de context zoals je weet is het aan jou om jouw deskundigheid en inzet en kracht, maar ook persoonlijkheid in te zetten.

Interviewer: want dat is het geloof dat als je meer autonomie geeft aan iemand dat er dan ook een betere uitkomst uit komt qua productiviteit, maar ook persoonlijk niveau dat er een verbetering wordt geboekt bij diegene die de beslissing neemt, klopt het dat dat een fundamentele aanname is van dat autonomie goed is?

EX1: de aanname die je mag doen is dat als mensen zich eigenaar voelen, percepieren als eigenaar van iets, of het nou een besluit is of een fysiek element dat er dan een andere verhouding ontstaat tussen datgene wat er moet gebeuren en de persoon zelf. Als het niet van mij is dan kan ik er ook afstand van nemen. Terwijl als het van mij is en ik er ook iets mee mag oden dan zit er een andere verhouding aan. Het mooiste voorbeeld is altijd de lease auto’s. De mate waarin mensen onzorgvuldigheid hebben ten aanzien van lease autos is vele male hoger dan wanneer mensen hun eigen auto hebben, want hij is niet van mij, er zit een contract overheen en verzekering en…

Interviewer: en dus veel meer krasjes

EX1: ja, onzorgvuldigheid

Interviewer: want dat is het geloof dat als je meer autonomie geeft aan iemand dat er dan ook een betere uitkomst uit komt qua productiviteit, maar ook persoonlijk niveau dat er een verbetering wordt geboekt bij diegene die de beslissing neemt, klopt het dat dat een
Appendix J: Excerpts from culture books and employee hand-books

Companies studied: Atlassian, Basecamp, Buffer, Disney, Facebook, Google, IDEO, Memoria Visual, Netflix, Nordstrom, Pandadoc, Spotify, Stripe, Thoughtbot, Valve,

General

“Technology is transforming virtually every business sector (...). As a result, barriers to entry that have stood for decades are melting away. Every incumbent business is vulnerable to competition and disruption. This transformation is happening at an unprecedented pace, and it’s accelerating. (...) Power has shifted from companies to consumers. (...) Meanwhile, within companies the power has shifted as well. Individuals and small teams can have a massive impact. They can create new ideas, experiment, fail, and try again, and get their successes to a global market.” (Google)

“Innovation can’t be owned or ordained it needs to be allowed. You can’t tell innovative people to be innovative, but you can let them” (Google)

“Culture is the manifestation of the shared value of the organization as represented by the actions of its members” (Spotify)

“What culture is not: Free food, slides in your office, photos with Daleks, concierge services, rock-climbing walls” (Spotify)

On culture

Good culture

What makes a good engineering culture? Stuff gets done, and done well. People are happy. Leaders provide direction and guidance and get out of the way; Stuff gets done. Success is celebrated. Failure is used as a way to learn; celebrate failure with the fail wall (whiteboard with post-its) (Spotify)

Protecting culture

Protecting your culture: Walk the talk, Hiring is crucial, Firing is also crucial, Communicate your values, Measure against your values, Your organization reflects your values, Watch out for warning signs (Spotify)

Actioning culture

“Real company values as opposed to the nice-sounding values, are shown by who gets rewarded, promoted or let go. Real company values are the behaviors and skills that we particularly value in fellow employees.” (Netflix)

Servant leadership: at Spotify managers are focused on coaching, mentorship and solving impediments rather than telling people what to do
**On strategy**

“A good culture isn’t the same things as a good business though. Still need a vision, a product and customers” (Spotify)

“It’s best to work in small teams, keep them crowded and foster serendipitous connections. Organize the company around the people whose impact is the greatest. Next comes strategy. (...) Smart creatives know [that any business plan is likely to quickly become obsolete], and will be scared that a business plan will hamper their freedom. Don’t base your venture on a plan, instead base it on a strategic foundation. You can have a plan, but know that it will change probably a lot. The plan is fluid, the foundation stable.” (Google)

“6 months or 30 years.

There is no point in having a 5-year plan in this industry. With each step forward, the landscape you’re walking on changes. So we have a pretty good idea of where we want to be in six months, and where we want to be in 30 years. And every six months, we take another look at where we want to be in 30 years to plan out the next six months.

It’s a little bit shortsighted and a little bit not. But any other approach guarantees everything you release is already obsolete.” (Facebook)

“Everyone has to function as a ‘strategist‘, which really means figuring out how to do what’s right for our customers” (Valve)

**On people**

“The people that can have the biggest impact of all are the ones we call Smart Creatives. These are the product folks who combine technical knowledge, business expertise and creativity. When you put today’s technology tools in their hands and give them lots of freedom they can do amazing things, amazingly fast.” (Google)

“Opportunity is everywhere. Smart creatives are everywhere. Ambitious people who want to build a team of the latter to pursue the former are everywhere” (Google)


“Put most simply, IDEO is all about our talent: finding, supporting, keeping, growing and nurturing them, not to mention, inspiring, supporting and enabling them once they’re inside. That’s you, or hopefully you, if you’re reading this - it’s now official: you are literally the center of our universe.” (IDEO)

**On teamwork**

“We’re a team, not a family. We’re like a pro sports team, not a kid’s recreational team. Coaches’ job at every level of Netflix to hire, develop and cut smartly, so we have stars in every position” (Netflix)

“Brilliant jerks; some companies tolerate them, for us, the cost to teamwork is too high.
Diverse styles are fine, as long as someone embodies the nine values” (Netflix)

**On communication**

“Communication is as important as decision making, and like decision-making it is something that most leaders think they are good at. They are mostly wrong.” (Google)

“Most forward-thinking companies (...) fail to understand what consensus means; It’s not about everyone agreeing, it’s about everyone being heard and then rallying around the best answer” (Google)

“By now it’s obvious that roles at Valve are fluid. Traditionally at Valve, nobody has an actual title. This is by design, to remove organizational constraints. Instead we have things we call ourselves, for convenience.” (Valve)

**On organisation**

“We love our Venn diagrams at IDEO and like many aspects of our work, the sweet spot of course lies in the middle. Enough design, but not too much. Enough structure without feeling too structured. Enough intention without feeling too controlling. We want to create enough of a system for us to function in but not so much that we feel, well, too functional.” (IDEO)

“We are creative people, so doing things that feel buttoned up and, well, organizational, doesn’t come easy to many of us.” (IDEO)

“there is no org-chart in the sky, no burning bush, master document, no big annual PowerPoint. We fall more into the campfire tradition – stories told, ideas shared, hopes and dreams recognized, and provocations from round the world gathered. The State of IDEO is as close as we will ever get to an “official” document, and I am happy to report that that is unlikely to change.” (IDEO)
Organisational structure at Spotify

“Highly aligned, loosely coupled.

Highly aligned. Strategy and goals are clear, specific and broadly understood. Team interactions are on strategy and goals rather than tactics. Requires large investment in management time to be transparent and articulate and perceptive and open.

Loosely coupled. Minimal cross-functional meetings except to get aligned on goals and strategy. Trust between groups on tactics without previewing/approving each one. Leaders reaching out pro-actively for ad-hoc coordination and perspective as appropriate. Occasional post-mortems on tactics necessary to increase alignment.”

“IDEO’s business and culture are complex, so the need for leadership at IDEO is both broad and deep, encompassing locations, skill areas, people, projects and clients, and it has many vectors and dimension that span the formal to the informal, so it is a challenging thing to manage.” (IDEO)

Values and principles

“Before anything else, values come first. Without clear, shared values, we wander independently and contradict one another. Everything’s harder when we all believe different things about what’s important to us, our company.” (Basecamp)

“valuable contributions internally transcend team borders.” (Stripe, 2018)

Calm

Levelheadedness (Basecamp) Basecamp: We should be calm, considered and thoughtful. (...) we don’t act out of spite, we don’t rush to judgment, we don’t jump to conclusions.

Collaborate

This is the one we always get asked about because people can’t believe it and it sounds too good to be true, and ironically it’s also the one that has the simplest and most powerful answer. We’ve all sat in studio lunches where work has been presented, and realized at some point that nobody has actually used the word “I,” right? Instead, every single
person, without exception, goes out of their way to acknowledge everyone around them, using the word “we”. We can be accused of a lot of things here, but not wholeheartedly supporting one another through the complex emotional minefield of our client projects is absolutely not one of them. Our most powerful asset in the arsenal, by far – the word ‘we.’ (IDEO)

“Because Stripe is highly interdependent, really good Stripes have a strong sense of overall ownership of the whole company but are non-territorial regarding their nominal domains. There are no bonus points for building large teams.” (Stripe, 2018)

Do

Talk less, do more (IDEO). IDEO: “Nothing is a bigger buzz-kill than over-intellectualizing, theorizing and postulating when drawing, making and building can tell a much more powerful story. So don’t be a theorist, be a pragmatist.”


Embrace ambiguity

Embrace ambiguity (IDEO) IDEO: “Imagine you and your teammates are in a rowing boat, on a lake, and it’s foggy. You need to collectively row together, not knowing when you will hit land, but knowing that eventually you will. Your job is to support one another, not jump overboard but pull together in one direction.”

Fall fast

“The quick shall inherit the earth” (Facebook)

“Fast is better than slow. While slow is adding unnecessary embellishments, fast is out in the world. And that means fast can learn from experience while slow can only theorize. Those who ship quickly can improve quickly

So fast doesn’t just win the race. It gets a head start for the next one.” (Facebook)

Feedback

“Expect to routinely contribute to projects across the company. Expect to receive feedback from engaged coworkers who have less state about your projects than you do” (Stripe, 2018)

“Everyone is a designer. Everyone can question each other’s work.” (Valve)

Freedom

“Our model is to increase employee freedom as we grow, rather than limit it, to continue to attract and nourish innovative people, so we have better chance of long-term continued successs.” (Netflix)

“Most companies curtail freedom as they get bigger. As they grow to avoid errors (sounds pretty good to avoid errors).

Desire for bigger positive impact creates growth. Growth increases complexity. Grown
shrinks talent density in most firms. Chaos emerges:

Process emerges to stop the chaos. Procedures; no one loves process, but feels good compared to the pain of chaos. Process-focus drives more talent out. Another option: 1) avoid chaos as you grow with ever more high performance people, not with rules. 2) then you can continue to run informally with self-discipline and avoid chaos. 3) the run informally part is what enables and attracts creativity.”

"Like ‘free speech’ there are some limited exceptions to ‘freedom at work’. Prevent irrevocable disaster. Morla, ethical and legal issues.” (Netflix)

“Rule creep, we try to get rid of rules when we can, to reinforce the point” (Netflix)

**Impact**

Impact: Accomplish amazing amounts of important work. Consistent performance demonstration. Focus on great results rather than on process. Exhibit bias-to-action and avoid. Analysis-paralysis. (Netflix)

**Independence**

Independence (Basecamp) – Basecamp: we encourage independent though and original thinking.

**Judgement**

“Our employee handbook is a single card that says ‘Use good judgment in all situations,”

(Nordstrom)

Be fair and do the right thing (Basecamp) Basecamp: we all have to use our best judgement (...) a good rule of thumb is "what would you do for a friend or a neighbor if they asked for help?"

Do the right thing (Buffer)

Judgment; Wise decisions despite ambiguity. Identify root causes, get beyond treating symptoms. Think strategically, and can articulate what your are and what you are not trying to do. Smartly separate what must be done well now, and what can be improved later. (Netflix)

Honesty: Candor and direct. Non-political. Admit mistakes. (Netflix)

**Learn**

IDEO: “Let’s be honest, what we do is really hard, we are constantly going into uncharted territory and if we were not trying new things that failed occasionally, we wouldn’t still be in business. When it happens to you, (which it will), own up, take a deep breath, have a glass of wine with your team and try to figure out what you have all learned and how to help others learn from it, so that we can all, well, learn together.”

Learn from failure (IDEO) (Buffer) (Spotify)

Buffer: Make time to reflect.
Spotify: Learn from failure. Innovation at every level.

Curiosity: Learn rapidly. Seek to understand out strategy, market, subscribers and suppliers. Knowledgeable. Contribute outside of your specialty. (Netflix)

**Listen**

“When criticized, we try to seek the truth in the accusation rather than activating our defensive shields. We invite people who many of us disagree with to come speak at Stripe and we welcome views that don’t obviously mesh with our own.” (Stripe, 2018)

Listen first, then listen more (Buffer)

**Open**

Default to transparency (Buffer)

Transparency, trust, servant leadership (Spotify)

**Optimism**

“Internally, we’re always thinking about what’s broken, which problems could lie around the corner, and where the unaddressed risks lie. But an important aspect of Stripe culture is macro optimism. We believe that Stripe will be far better in the future than it is today. When considering ideas, we think “how might it work?” is more interesting than “why will it fail?”” (Stripe, 2018)

One thing that excites us all is the collective excitement that we all have for well, pretty much everything. IDEO is a place where possibility meets tangibility, and that is what our clients are asking of us – they are asking us to be optimistic about their future, believe, as optimists do, that doing good begets doing good, and believing that something is possible will somehow make it so. And most times it does. One of the best business quotes of all time is from Jay Chapman, the CHO (Chief Happiness Officer) of Prêt a Mangér, the British sandwich chain, who famously said: “You can’t hire someone who can make sandwiches and teach them to be happy. So we hire happy people and teach them to make sandwiches.” We are extremely lucky at IDEO: we have both happy people and we happen to make great sandwiches. (IDEO)

**Ownership**

It’s kind of interesting how this follows collaboration, and how on some level they sit in tension. But collective interdependence also needs individual independence to underpin it. By which we mean: take responsibility for your bit, own it, and resolutely follow through on it. So the unwritten social contract here is that individual ownership supports collective responsibility. Own that. (IDEO)

“You own your work! Yep! Who want swork to be line an angry skunk we’re eager to run away from? Take ownership on goals, quality and neatness, on making it happen and the well-being of your work. Individual ownership makes the collective shine bright.” (MV)

**Passion**

Selflessness: Seek what is best for Netflix. Ego-less. Help others. Share information openly. (Netflix)

Personal development

Focus on self-improvement (Buffer)

“We want to support every employee in building their own personal brand. After a few years at PandaDoc your LinkedIn should look like you’re an absolute superstar, guru, ninja, industry mogul, etc. Here are a few ways we can make that happen: 1) Google “Mikita Mikado” 2) Google “Jared Fuller” 3) Speak at conferences and events, as long as you’re speaking in front of our target buyers or prospective employees.” (Pandadoc)

Positive

Choose positivity (Buffer) (IDEO). IDEO: Be optimistic. “You can’t hire someone who can make sandwiches and teach them to be happy. So we hire happy people and teach them to make sandwiches.”

Straightforward

Basecamp: speak plainly and clearly, watch out for lingo, assumptions, exaggeration or other things that get in the way of straightforward explanation.

Communicate with clarity (Buffer)

Communication: Listen well. Concise and articulate. Respectful. Calm. (Netflix)

Courage: Speak you mind. Tough decision making. Smart risking. Question actions that are inconsistent with our values. (Netflix)

Support

“We want to work in a company of deeply good people who treat their colleagues exceptionally well. No matter how talented, we won’t hire jerks.” (Stripe, 2018)

Generosity (Basecamp) Basecamp: being generous is surprising someone on the other end with goodwill and asking nothing in return

Make others successful (IDEO)

Trust

“no matter how strong the disagreement, we believe firmly in the importance of trusting each other’s intentions.” (Stripe, 2018)

Other

Be a “no ego” doer (Buffer)

Live smarter, not harder (Buffer)

Show gratitude (Buffer)

Iterative development, agile-first, data-driven autonomous teams (Spotify) Continuous
improvement, shared responsibility (Spotify)

Ways of working

Basecamp method:

6 week cycle. “this fixed cadence serves to give us an internal sense of urgency (...) the is is not that everything we ever decide to work on has to take six weeks or can be completed in that time. But rather that we think about how we can break big projects into smaller ones (...)”

Small batch. “We work on stuff that won’t take longer than 2 weeks at the maximum”

“The general concept of the cycle extends to all departments. It gives us a regular rhythm to heartbeat on, and it allows us the patience to keep messing with the priorities for at least six weeks at the time.”

Heartbeats: “we have three chief mechanisms for keeping everyone in the loop about the work that’s going on”:

• “What did you work on today?” – “They’re a great conversation starter if you see someone working on something you either care about or want to learn more about. Please do use them as such!”

• “What will you be working on this week?” -

• Team updates at halfway through a cycle, at end of cycle or when something new is launched. This is where the big presentation of work is done, and the main way for you to keep in the loop with what the company is focused on at a high digestible level.

Pitches. “Write-up your idea of a new feature, a change to a feature, or any other product development you think we should be considering as a fully considered post (the more specific, the better). This gives the whole company a chance to consider and respond to the idea, and then we’ll have the idea encapsulated in a post, available for reference at any time.”

“It’s far better for everyone’s concentration and sanity if you collaborate as though most things will get an answer eventually, but not necessarily right this second.” (Basecamp)

“Organizational theory is thick with descriptions of the trade-offs between functional and project company structures. We seek to be more project than functional. This means a single project team should be able to go from idea to deploy as independently as possible. (...) s soon as organizational bottlenecks form, like a slew of features waiting for “the mobile integration”, we’re dragged towards more micro and detailed schedule management. It becomes a critical path with dependencies and making sure team Z is available just at the right moment for team A, such that nobody is blocked. That’s a poor fit for our organizational aspirations, so we have to work to counter that.

Rituals at Basecamp:

• Twice a year company meetup in Chicago for a week (loosely scheduled)
  • Every morning breakfast at 8am
• Sessions at 9am
  • Monday or Tuesday morning: CEO update about company
• Lunch shared in the office
• Dinner up to you
  • All-company dinner on Tuesday evening
• Special sessions:
  • Recognition awards presentation
    • Nominated employees who make Basecamp a better place to work.
    • 5 winners each meet-up
    • Person who nominated the other does the presenting
    • Price: wood thing handmade, each edition a different kind of wood
• Everyone On Support; everyone is called upon to do a support shift every month or two
• 5x12s; monthly randomly select 5 people to chat on a call with Jason and David (leadership)

What others say

“Do work you’re proud of. Well, well, this plumbing is leaking, a bit of chewing gum will sort that out! There is no excuse for shoddy or “that will do the trick!” work. Even the Enterprise needs engine maintenance, and there will be the odd bug to squash. The seeds of greatness are a collection of finely executed details, so let’s make it spotless and take pride in everything we do.” (MV)

Practicalities

“The word ‘asenteism’ has been officially drummed out of the Disney vocabulary. It sounds chronic, incurable and has been considerably over-worked.” (Disney)

“We should focus on what people get done, not how many hours or days worked. Just as we don’t have an 9-5 day policy, we don’t need a vacation policy.” (Netflix)

On-boarding

“Getting started at Basecamp can be overwhelming. There are a lot of little details, a number of big tasks, you’re learning a new job and new coworkers, and you’re likely doing all these things while working remotely. Your Basecamp buddy, your Ops buddy, and your manager are all here to help. If you don’t know how to set up your machine (or why you’re setting up something on your machine!), please ask. There is no set time period for most training periods, but most people are up to speed in about 3 months.” (Basecamp)

“We expect a lot of autonomy from Stripes both in the work they do and in their own development. We believe in performance management and feedback, but we’re not rigid in terms of a career paths and box checking” (Stripe, 2018)

“What we work will mean some late nights, some weekends, and (especially if you end up in a position of significant responsibility) paying attention to email even during off-hours. (...) Our business is intertwined with the global economy, so while Stripes take holidays, Stripe does not” (Stripe, 2018)
“You will also be surrounded by exceptionally motivated, driven people. (...) you’ll almost never be annoyed about that slacker in the next cubicle (and not just because we don’t have cubicles). But it can also be stressful: if you compare yourself to others, you will almost always see someone working harder, staying longer, or being more successful.” (Stripe, 2018)

“It’s also worth mentioning that joining Basecamp can be all-consuming. We’ve seen it happen. You dig Basecamp, so you feel pressure to contribute, maybe overwhelmingly so. The people who work here are some of the best and brightest in our industry, so the self-imposed burden to be exceptional is real. But here’s the thing: stop it. Settle in. We’re glad you love this job because we all do too, but at the end of the day it’s a job. Do your best work, collaborate with your team, write, read, learn, and then turn off your computer and play with your dog. We’ll all be better for it.” (Basecamp)

“We’re all working online some nights and weekends, responding to emails at odd hours, and taking an afternoon now and then for personal time” (Netflix)

Career

“Advancing your career at Basecamp doesn’t mean giving up on your craft and moving into “management”. Whether you work in programming, design, ops, support, or whatever, you can become better at the work itself and level-up that way.” (Basecamp)

“Individuals should manage their own career paths, and not rely on a corporation for planning their careers” (Netflix)

“To avoid surprises, you should periodically ask your manager; “if I told you I were leaving, how hard would you work to change my mind to stay at Netflix?” (Netflix)

The keeper test managers use: ‘which of my people, if they told me that were leaving in two months for a similar. Job at a peer company, would I fight hard to keep at Netflix?” (Netflix)

Advice for managers

“Managers; when one of your talented people does something dumb, don’t blame them. Instead, ask yourself what context you failed to set. (...) when you are tempted to ‘control’ your people, ask yourself what context you could set instead. Are you articulate and inspiring enough about goals and strategies?” (Netflix)

“If manager would promote employee to keep them if employee were thinking of leaving, manager should promote now, and not wait.” (Netflix)

Motivation

“You will, within weeks of joining Stripe, work on problems that no one here has solved before. (And you’ll occasionally end up working on problems that no one anywhere has solved before.)” (Stripe, 2018)
Performance monitoring

“We perform bi-yearly reviews in May and November. Everyone who has been with the company at least 90 days gets a review. The primary purpose of these reviews is to give feedback on career path advancement and recognize accomplishments. We wrap up our November reviews in time to make promotion related pay adjustments effective January 1st.” (Basecamp)

Peer reviews (Valve)

Stack ranking (and compensation) (Valve) – rank each other against our peers, in order to gain insight into who’s providing the most value at the company and to thereby adjust each person’s compensation to be commensurate with his or her actual value.

Inspiration

“We’re not a very competitive culture in the sense that someone else does not need to lose for you to win” (Stripe, 2018)

“Our culture is a work in progress. Every year we try to refine our culture further as we learn more.” “Success at Stripe means seeking out the ski slopes that are just a bit too steep” (Stripe, 2018)

“Everybody want to fly to the moon or mars. Boring. Not us, we’re aiming for pluto.” (MV)

“Our employee handbook is a single card that says ‘Use good judgment in all situations,’” (Nordstrom)

“Try to imagine the unimaginable, because unimaginable things are happening a lot. Then make a bet on that future.” (Google)

“Ask yourself, what could be true in 5 years?” (Google)

“By the standards of the rest of the world, we overtrust. We’re okay with that.” (Stripe, 2018)

“no matter how strong the disagreement, we believe firmly in the importance of trusting each other’s intentions.” (Stripe, 2018)

“When you don’t realize what you can’t do, you can do some pretty cool stuff.” (Facebook)

More inspiration

Inspiration: https://github.com/basecamp/handbook/blob/master/what-influenced-us.md

Organisational chart: https://github.com/basecamp/handbook/blob/master/orgchart.md

Guidelines for side-gigs: https://github.com/basecamp/handbook/blob/master/moonlighting.md
Amazing toolset: https://www.atlassian.com/team-playbook/plays?painPoints=empathy-debt

Open-sourcing value finding with surveys throughout the company: https://open.buffer.com/creating-values/


Bibliography

Atlassian: https://www.atlassian.com/team-playbook

Basecamp: https://github.com/basecamp/handbook

Buffer: https://open.buffer.com/creating-values/


Google: https://www.slideshare.net/ericschmidt/how-google-works-final-1

IDEO: https://lboi.ideo.com


Pandadoc: https://www.pandadoc.com/pandadoc-culture-code/

Spotify: https://www.slideshare.net/kevingoldsmith/building-a-strong-engineering-culture-my-talk-from-bbc-develop-2013

Stripe: https://stripe.com/us/jobs/candidate-info?a=1#culture

Thoughtbot: https://thoughtbot.com/playbook


Not yet: Hubspot
Appendix K: Momento - mini-booklet
**how to use this card-set**

Momento is a way of working for ODS and its teams. This card set is made specifically for teams. It is a pocket-sized explanation of the way of working and includes a miniature version of all the tools necessary to start working with Momento.

This card-set is split in 2 halves. The first half introduces the basic principles of the method and its meetings. The second half contains the Momento miniature toolkit. This half contains 2 miniature versions of the canvas tools used by the teams, as well as a set of cards that outline the purpose, owners, and agenda of each meeting.

**Colophon**

Stijn Bakker
stijn.bakker@skies.com

![KLM Royal Dutch Airlines](logo.png)

![BCG](logo.png)
**momento**

**momento is a way of working**
It consists of two circles:
- alignment meetings
- sprints

**at the core of momento lies goal-setting**
Multiple goals, in fact:
- one for 3 months
- one for 1 month
- and for the sprint

**a clear goal is the core of any team**
this card-set supports goal-setting
backed up by a method
to turn translate those goals into action
the method

PI events

evry three months a PI event is organised
this is where the team sets a 3-month goal
user boards

every month the product owner presents the latest product in the User Board.

This is where the product owner gathers feedback from stakeholders.

sprints

The stakeholder feedback and 3-month goal are the input for the sprints.

Sprints are smaller cycles of 2 weeks in which a product is developed.
the sprint
consists of 5 different meetings
each with their own purpose, agenda and owner

sprint retro: evaluate learning
sprint review: a moment to gather all feedback and set goals
sprint planning: turning goals into a concrete to-do-list
sprint stand-up: daily moment to quickly evaluate progress
sprint demo: gather user feedback with a product demonstration

sprint review
during the sprint review the team creates an overview of user feedback, stakeholder feedback, technical debt and the ODS mission, using the statement of direction canvas (see momento miniature booklet later on in this card-set). Using this canvas the team defines a 3 month, 1 month and/or sprint goal
sprint retro

During the sprint retro the team evaluates the performance of the team in the last sprint, using the retrospective canvas (see momento miniature toolset, later on in this card set). At the end of the retro meeting the team sets an area to focus on in terms of team performance during the coming sprint.

sprint planning

During the sprint planning the team takes the team focus (defined in sprint retro) and sprint goal (defined in sprint review) and translates these goals into an action plan laying out the actual to-do-terms and assignment of these items to team members. For this, the team uses the online tool Jira to create and assign tickets.
sprint stand-up

During the sprint the team has a daily stand-up with only the team-members to give an update on progress. During the meeting each member presents 1) what he did yesterday, 2) what he plans on doing today and 3) whether he foresees impediments. The sprint stand-up takes maximum of 15 minutes, and should focus on giving an update, not discussing progress. To oversee this someone of the team is dedicated to be stand-up boss. It is his/her job to monitor time and ‘step in’ during discussions. The stand-up boss gets a set of cards to manage the standup (see moments: miniature toolset).

sprint demo

During the sprint demo the team presents a new version of the product to (a selection of) users and stakeholders. Each new feature is presented and outlined, and after each feature the users are asked for their feedback. It is the task of one dedicated team-member to make notes about the feedback from users per feature. This feedback will later be collected in the sprint review. The sprint review is also a moment to celebrate progress of the team.
momento
miniature toolset

statement of direction
canvas

- Statement of direction - canvas
- Retrospective - canvas
- Meeting cards

Statement of direction canvas is a tool that the team can use during sprint reviews and PI events to set a clear goal. It consists of two halves. In the left side of the canvas, the team collects feedback from users, stakeholders, restates the ODS mission and notes technical debt that needs to be resolved. With this overview of information, the team can start to formulate a clear goal for the product in a single sentence. The right side of the canvas supports the team to do so through a sentence with blanks to be filled in.
Statement of direction
For (users)
during (a situation/use case),
(product) produces (main outcome)
by (presenting something to the user),
determined by (levers/working principles),
taking into account
(limitations of the working principles)
to (solve a problem).

Example 1:
For flight controllers in LU during everyday preparation for next day by assigning flights to tails, Pathfinder produces an optimized list of tail-assignments through swap, multi-swap rotations within subtypes taking into account tail-rules, flight-links, unassigned maintenance slots and delay prediction presented to the user in an altered flight schedule and exportable flash-schedule file to support flight controllers in setting up a robust schedule.

user input: feedback from users, gathered through sprint demo meetings and co-creation with users

technical input: an overview of technical debt, gathered during development, captured as separate tickets in Jira

stakeholder input: main stakeholders and their needs and feedback, gathered through user board meetings by the Product Owner, and informal conversations with stakeholders

strategic input: restatement of original mission of ODS and the team
**FRONT**

**retrospective canvas**

<table>
<thead>
<tr>
<th>satisfaction + average</th>
<th>what went well</th>
<th>what could be improved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**BACK**

Retrospective canvas is a tool that the team can use during sprint retro to structure the team evaluation and set a teaming goal. It consists of five squares to be filled in. Each member writes his satisfaction with the last sprint (scale 1-10), and the scrum-master calculates and notes the average. Individually each team-member presents what went well and what could be improved. During this presentation, the team brainstorms on action items to improve (to-do’s). Finally, the team picks one teaming goal or action item for the team to focus on in the next sprint.
Appendix L: Momento - Little book of ODS
the little book of

ODS

Royal Dutch Airlines

BCG
THE BOSTON CONSULTING GROUP
we believe in human creativity and the logic of algorithms

we build a world for the two to strengthen each other
our mission:

create actual intelligence
intelligence.

the ability to make decisions

We define intelligence as the ability to make decisions, and to make them well. A decision consist of choosing an action from a set of available actions. We call a decision good when the action is chosen and done by the right person, at the right time, with the right information, with the maximum likelihood of achieving the best outcomes.
imagine you’re a gate agent.

You make sure passengers get onto their plane safely, on time and hopefully with a smile, no matter the circumstances. And today, the circumstances start out pretty good: the sun is shining, it’s not too early in the morning and it looks like all your passengers are at the gate. After a quick check with the team leader if the plane is ready, you board the passengers and send them off to the bus that will take them to their airplane. As the bus leaves for the plane, you allow yourself a small amount of satisfaction: you’re like a shepherd, a good, caring shepherd, that sent her flock off to wherever they might be going. Nice job.

But then you get a beep and...ugh. The plane is sending the passengers back due to a technical glitch which they’re still waiting to repair. Alright - you can handle this. Of course, it’s frustrating that they apparently sent the wrong message earlier, but no use crying over split milk, right? You set yourself to the task of deboarding all passengers, check the expected delay and communicate clearly to your passengers: the delay is expected to last for three hours, but it’s always possible that after repairs the flight gets bumped up, meaning that you could also leave within half an hour. Don’t stray away too far from the gate and always keep your eye on the information screens, please.

And indeed, within half an hour you do get the go-ahead to board again. Which you dutifully announce and commence immediately. Good, you see some passengers even running towards the gate, awkwardly juggling their hand luggage and boarding passes. But a few minutes later you’re less amused: three passengers haven’t returned to gate, meaning you can’t leave. A tense waiting game now commences: you send out announcements and hope the passengers hear it over the pints of lager you expect them to be drinking in one of the airport bars, before you have to make the awkward decision to offload their luggage. You try to stall as much as possible but you also know there are other flights supposed to be docking at this gate, other crew members needing to be somewhere else, and then you breathe a sigh of relief: in the distance, you see the three missing passengers scuttling towards you. Yes, by the looks of it, they indeed managed to down a few beers in the short period they left from under your caring gaze, but no matter. You prepare to make a light-hearted joke about it whilst they’re boarding, when you receive the dreadful message: the wait has been too long, and OCC has decided to cancel the flight entirely...

What will you do now?
or, imagine you are a flight controller at Operations Control Centre.

Your daily job is to schedule and reschedule: in front of your computer screen you see a chart showing all of tomorrow’s flights. Your screen looks a bit like little lego blocks, or a tilted version of tetris. The left side of each block is supposed to represent the arrival of your flights, the right side their departure, and the length in between shows how long this plane is supposed to stay on the ground. You say ‘supposed to’ because these blocks keep changing all the time.

Like this morning: the first piece of news you receive after you fired up your computer, is that the incoming flight from Atlanta is expected to arrive two hours later and needs some additional maintenance. Nothing serious, they just encountered a flock of hapless birds and a thunderstorm. You check your screen and try to gauge the consequences: if this flight is two hours late and needs extra maintenance, what are the consequences for the crew? Can they still make their next planned flight? Is the maintenance crew available?

You’re about to turn to your colleague about the maintenance crew planning, when you receive another message: there’s a storm expected tomorrow. Holy smokes! That’s a wholly different ball game! A lot of flights will probably be grounded or delayed. A hefty dose of adrenaline gears you up for what’s going to be a pretty intense day when finally... the screen freezes. The software blacks out...

What will you do now?
these are the people ODS helps:

people facing complex decisions
how we help
the right information, at the right time, available to the right person

Take the example of the gate agent above: she now has a herd of passengers, who each are very disappointed and need to be...to be what exactly? If the gate agent at that exact moment in time has access to all the relevant information of all the passengers, and all the relevant information about options available to her, she can choose better what to do with each and every one of them.

Likewise for a landing site agent who has to decide what to do with all the pieces of luggage that a plane spits out after landing: if she knows where each piece of luggage needs to go, she will be better able to distribute it.

Making information like this available to the right person at the right time is a matter of data science and design: a data scientist tries to figure out where the data is and provide insights into it, and a designer finds ways to present this effectively and attractively to the person who needs to use it.
choosing actions
with the maximum likelihood
of best outcomes

Take the flight controller, deciding what to do with a plane that’s been delayed: should he swap the landing window of this plane with that of another? This requires data science again: we need to understand the dependencies and feasibilities that each option brings. Crucially, many of these dependencies are risky or uncertain. For example, you get a weather report stating there will be thunderstorms next week. These thunderstorms could obviously influence traffic patterns at the hub with planes coming in late and being grounded. But how sure are you that the thunderstorms will actually come? Can you meaningfully assign probabilities to this prediction or not? Crucially, there’s also an uncertain relation between actions and outcomes: you might assume that deciding to closing the gate five minutes later will lead to the plane departing five minutes later, but you can’t be sure. Thinking about relations like this is a job for prediction modelling, which has many tools in its toolbox but lately has been using machine learning an incredible lot.
But then, assuming you’ve charted all relations between actions and their consequences, and know which ones are certain, which ones are risky and which ones are uncertain, than you’re still not done. This is the part where you maximize for the best outcome. Humans do this ‘by eye’ and ‘by experience’: experience tell the flight controllers that swapping plane A and plane B might be a good idea.

We do this by intuition because decisions like this are complex: they involve a lot of variables. In jargon, they’re often NP-hard problems. It’s the job of optimizers to fight their way through these complex problems using the best weapon they have: optimization algorithms. Does swapping one plane for another on balance cost us more money than not swapping? We’ll only know for sure if we combine data science, prediction modelling and optimization techniques: only these three together will generate a clear recommendation. Generating this recommendation on time, such that the flight controller can take it into account, is a wholly different ball game again: it needs to be real-time, and the raw amount of computational power required to be able to do this, asks the most of our hardware and architecture.

Then, finally, the question is whether the recommendations are actually sensible. Having all our machines analyse the swapping possibilities in real-time might lead to the recommendation that swapping a plane that’s just about to taxi to the runway for take-off with a plane that’s still boarding, might make financial sense: it enables us to carry high-financial-value passengers a little bit faster to their destination than not-so-high-financial-value passengers. But haven’t we then optimized for the wrong goal (short term financial gain)? Shouldn’t we have optimized for both financial gain and maximum customer experience – and what’s the ideal balance between the two? Again, this is a question for optimizers, and a question that ODS relishes to answer.
This helps you understand a bit more about why the people at ODS do what they do. It also helps you understand a bit more what the purpose is of the tools we've built so far. Here they are:

**Blue Lagoon** - we call this our 'data lake': it’s the place where we store our historical and real-time data.

**Flight 720** - this is a tool that dips into our data lake and provides real-time data to everyone who is involved with the turn-around of an airplane, from gate agent to pilots and everyone else.

**Plug** - is the button that end users actually see on their iPads that presents Flight 720's data to them.

**Harbinger** - provides recommendations for plane swaps in the window of 30 days before operations to one day before operations. Calling it an incredibly smart scheduling tool means you’re hurting its feelings.

**Vanguard** - is Harbinger’s younger and faster sister. It provides recommendations for plane swaps on the one-day-before-operations window. Since probabilities of events, interrelated dependencies, and feasibility are whole lot more complex and time-pressure is a wee bit higher, this requires a separate scheduling tool... oh come on, now you're making Vanguard cry, too?

We’ve got more products, tools, platforms, or whatever you call them, like Sentry, Pathfinder, et cetera. But more important than knowing their name is knowing their place in the vast circus of decisions being made every day at an airline: which person do they help, at what time, with what information, maximizing for what outcome? It’s at the place and time of decision where the magic of intelligence happens - and that’s where ODS wants to be.
how we build
teaming

(with some top secret ingredients)
first and foremost, we’re a team

We’re on a mission: we’re making KLM operations data-driven. It’s a challenge, of course, and that’s why we have to stick together. Not as a family, as a team. Not just your own small team, but the entire ODS-team! See a new face? Come say hello. Ask questions, form a team. We’re in this together.
we organize in smaller teams

One giant team is not really practical, so each of you has its own smaller team. Each product team consists of experts (like consultants, data engineers, etc.), a product owner and a scrum-master. The scrum-master is sometimes part of multiple teams.

One of the teams is the management team (MT), arranging the strategy and resources of OOS. The product manager, part of the management team, forms a group with the product owners; the ‘product alignment’ group. We encourage all other experts also to form groups, to share knowledge and expertise.
top secret ingredient #2

our principles

be a team
Rule no. 1: we’re a team. We build together, on each other strengths. Only through collaboration can we transform KLM. So help each other out, go have drinks (though not too much), ask how it’s going.

own your work
We're building something pretty awesome, you can be proud of it. You build it. You. Nobody else. That means when it works, you're 100% allowed to celebrate success. But if it fails for some reason, also take your responsibility. Own your work.
fail right

We don't like failing. Failing feels awful. But failing is also necessary to learn. So if you're going to fail, better make it quick: fail fast. But this doesn't cover it all. We want you to fail right. That means going in with the idea that it is going to work, and a really specific objective of what you want to test. Making your goal concrete allows you to reflect, and learn. This is failing right.

just do it

Got an idea for a product, an improvement or maybe a fun activity? Go for it! Try to make it small, testable. do it. If you're in doubt whether an idea is worth trying,
ask your close colleagues first.
Moments is a way of working for ODS, and for its teams. The basic idea of moments is goal setting: a goal for the coming three months, a goal for the coming month, and a goal for the coming sprint. This allows us to be deliberate about where we’re heading, but also be flexible to new input we gather along the way, from users and stakeholders.

Three months, that’s the duration of our rhythm. It starts with a PI event, followed by 6 sprints. During the PI event the team sets a 3-month goal, 1 month goal, and sprint goal. Don’t worry, we’ve got a toolset for this. Then every month after a user board where the teams’ product owner presents the latest progress to stakeholders, the team creates a new 1-month goal, and subsequent sprint goal.

**sprint** (top secret ingredient M)
We develop in 2-week sprints, together with our users. It’s very similar to scrum, but a little different, read about it on the next page in more detail.

**PI event**
The start of the quarterly cycle. During the PI event every team defines its goal (we developed a separate toolset for you to dose ;) ) and presents and aligns this goal an plan with other teams.

**User board**
Every month we gather the most important stakeholders per business domain, and ask them for their feedback on our new products. This feedback is then used by the teams to create monthly goals.

**Demo**
Bi-weekly demo-meetings are an opportunity for ODS to come together and put the spotlight on one or two teams. These teams give a presentation on their progress, so that we all can catch up on what we’re making.

**Townhall**
ODS is pretty big, and we all have questions we want to ask to management. The townhall is an opportunity to do so, every 5 weeks, in a setting where everyone at once can hear the answers to these questions.
Sprint cycle

The sprint consists of 5 different meetings, each with their own purpose, agenda and owner.
**sprint review**
During the sprint review, the team creates an overview of user feedback, stakeholder feedback, technical debt, and the CDS mission, using the statement of direction-canvas (see moment miniature toolset). Using this canvas, the team defines a 3-month, 1-month, and/or sprint goal.

**sprint retro**
During the sprint retro, the team evaluates the performance of the team in the last sprint, using the retrospective canvas (see moment miniature toolset, later on in this card-set). At the end of the retro meeting, the team sets an area to focus on in terms of team performance during the coming sprint.

**sprint planning**
During the sprint planning, the team takes the team focus (defined in sprint retro) and sprint goal (defined in sprint review), and translates these goals into an action plan, laying out the actual to-do items and assignment of these items to team members. For this, the team uses the online tool Jira to create and assign tickets.

**sprint stand-up**
During the sprint, the team has a daily stand-up with only the team members to give an update on progress. During the meeting, each member presents: 1) what he did yesterday, 2) what he plans on doing today, and 3) whether he foresees impediments. The sprint stand-up takes a maximum of 15 minutes, and should focus on giving an update, not discussing progress.

**sprint demo**
During the sprint demo, the team presents a new version of the product (a selection of) users and stakeholders. Each new feature is presented and outlined, and after each feature, the users are asked for their feedback. It is the task of one dedicated team member to make notes about the feedback from users per feature. This feedback will later be collected in the sprint review. The sprint review is also a moment to celebrate progress of the team.
Appendix M: Statement of direction - canvas
# Statement of Direction - Canvas

<table>
<thead>
<tr>
<th>User Input</th>
<th>Technical Input</th>
<th>Version Number:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>For ____________ (users)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>during ____________ (a situation/use case)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>____________ (product) produces ____________ (main outcome)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>by ____________ (presenting something to the user)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>determined by ____________</td>
</tr>
<tr>
<td>Stakeholder Input</td>
<td>Strategic Input</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>____________ (viewers/working principles)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>taking into account ____________</td>
</tr>
<tr>
<td></td>
<td></td>
<td>____________ (limitations of the working principles)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to ____________ (solve a problem)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Delivery Deadline:</strong> ___ - ___ - _______</td>
</tr>
</tbody>
</table>
Appendix N: Retrospective - canvas
<table>
<thead>
<tr>
<th>satisfaction</th>
<th>what went well?</th>
<th>what could be improved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>average</td>
<td>action items</td>
<td>team focus for next sprint</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix O: Momento - card-set
**momento - overview**
consists of 5 different meetings
each with their own purpose, agenda and owner

**the sprint**
consists of 5 different meetings
each with their own purpose, agenda and owner

**sprint retro:** evaluate teaming
**sprint review:** a moment to gather all feedback and set goals
**sprint planning:** turning goals into a concrete to-do-list
**sprint stand-up:** daily moment to quickly evaluate progress
**sprint demo:** gather user feedback with a product demonstration

**Name:** PI event
**When:** Every 3 months
**What:** Moment to set a 3-month goal and align with other teams
**Tools:** "Statement-of-direction" canvas

**Name:** User bound
**When:** Every week
**What:** Progress meeting with stakeholders to gather stakeholder input
**Tools:** -

**Name:** Sprint
**When:** Every 2 weeks
**What:** Development method
**Tools:** Card set, retrospective canvas, "statement-of-direction" canvas
Statement of direction

For (users)
during (a situation/use case),
(product) produces (main outcome)
by (presenting something to the user),
determined by (levers/working principles),
taking into account
(limitations of the working principles)
to (solve a problem).
**Pi event**  
Every 3 months  

**Purpose**  
- set 3 month goal  
- align with other teams  

**Owner**  
PO  

**Preparation**  
- review user feedback  
- PO & Dev team  
- notes on technical debts  
- Jira (Dev team)  
- ideas on stakeholder input (PO)  

**Agenda**  
- set by PO  
- statement of direction canvases  

**User board**  
Every month  

**Purpose**  
- involve stakeholder feedback  
- gather stakeholder feedback  
- Product Manager & PO  
- Product presentation (PO)  
- collect stakeholder feedback (PO)  

**Agenda**  
- set by Product Manager  

**Tools**  
-  

**Back**
Sprint review

Every sprint

Purpose:
- Set 1 month goal
- Review completed work
- Next steps

Owner:
PO

Preparation:
- Notes on user feedback
- Notes on stakeholder feedback (PO)
- Statement of direction
- Drafting

Agenda:
- Recap progress
- Next steps
- Discussion
- Q&A
- Q&A
- Q&A
- Q&A
- Q&A
- Q&A
- Q&A
- Q&A

Task:
- "Statement of direction" canvas

Sprint retro

Every sprint

Purpose:
- Evaluate team performance

Scrum Master

Agenda:
- Satisfaction grade
- What went well
- What could be improved
- Next sprint
- Next sprint team focus

Task:
- Retrospective canvas
### Sprint planning

**Every sprint**

**Purpose**
- Define and assign tickets for upcoming sprint

**Owner**
- Scrum Master

**Preparation**
- Sprint team (from sprint retrospective)
- Sprint goal (from sprint review)
- Overview availability (overall calendar)

**Agenda**
- List team focus and sprint goal
- Define team availability
- Prioritize tickets
- Assign tickets

**Tools**
FRONT

REMINDER CARD

discuss later

STAND UP BOSS

for today

BACK
TWO Much CONTENT

Sprint demo
Every sprint
Purpose:
- Coordinate progress
- Gather user feedback

User:
- PO

Preparation:
- Deployment of new features
- Present feature feedback

Agenda:
- Invite users
- Present features
- Gather user feedback
- Repeat

Tools: