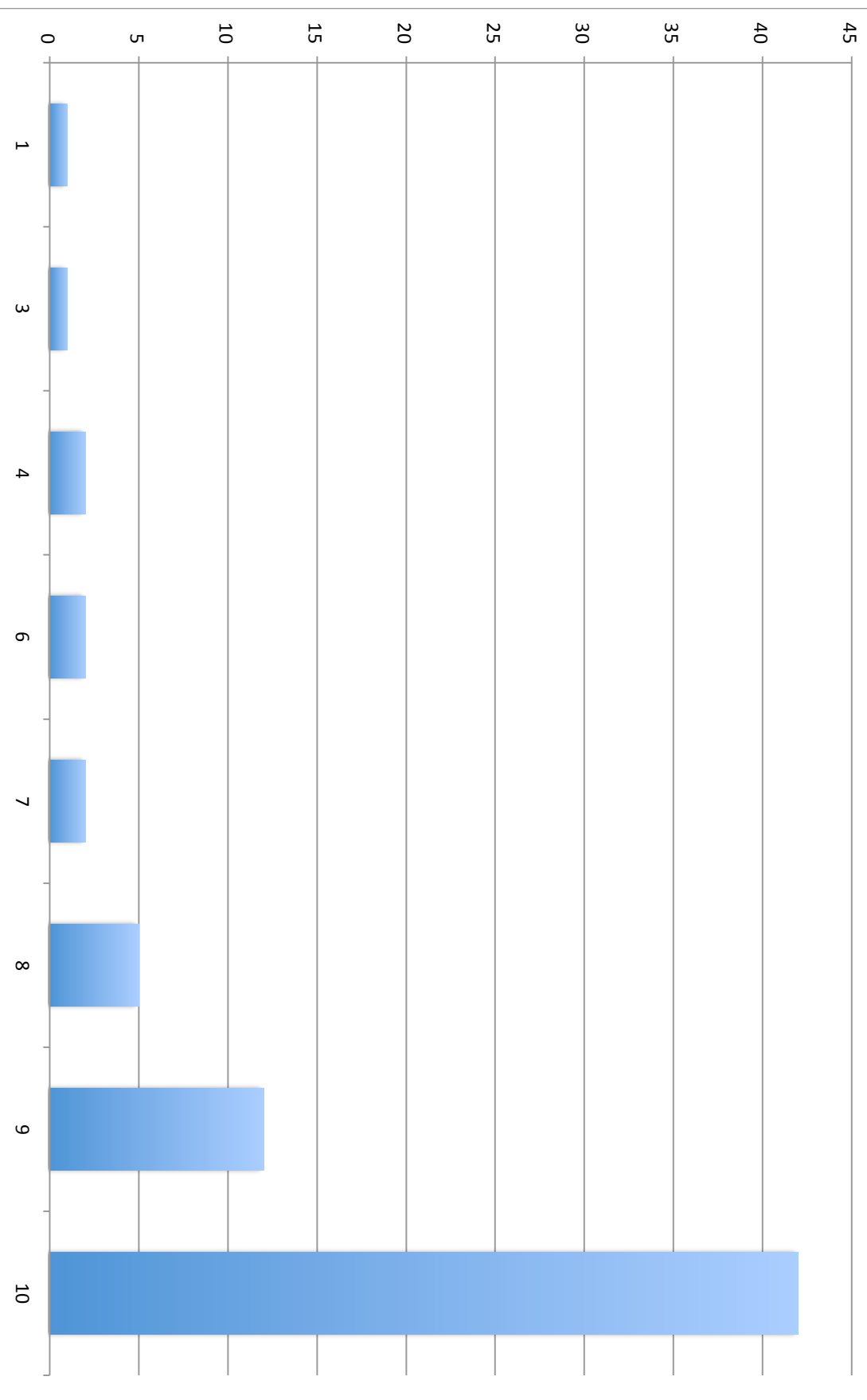
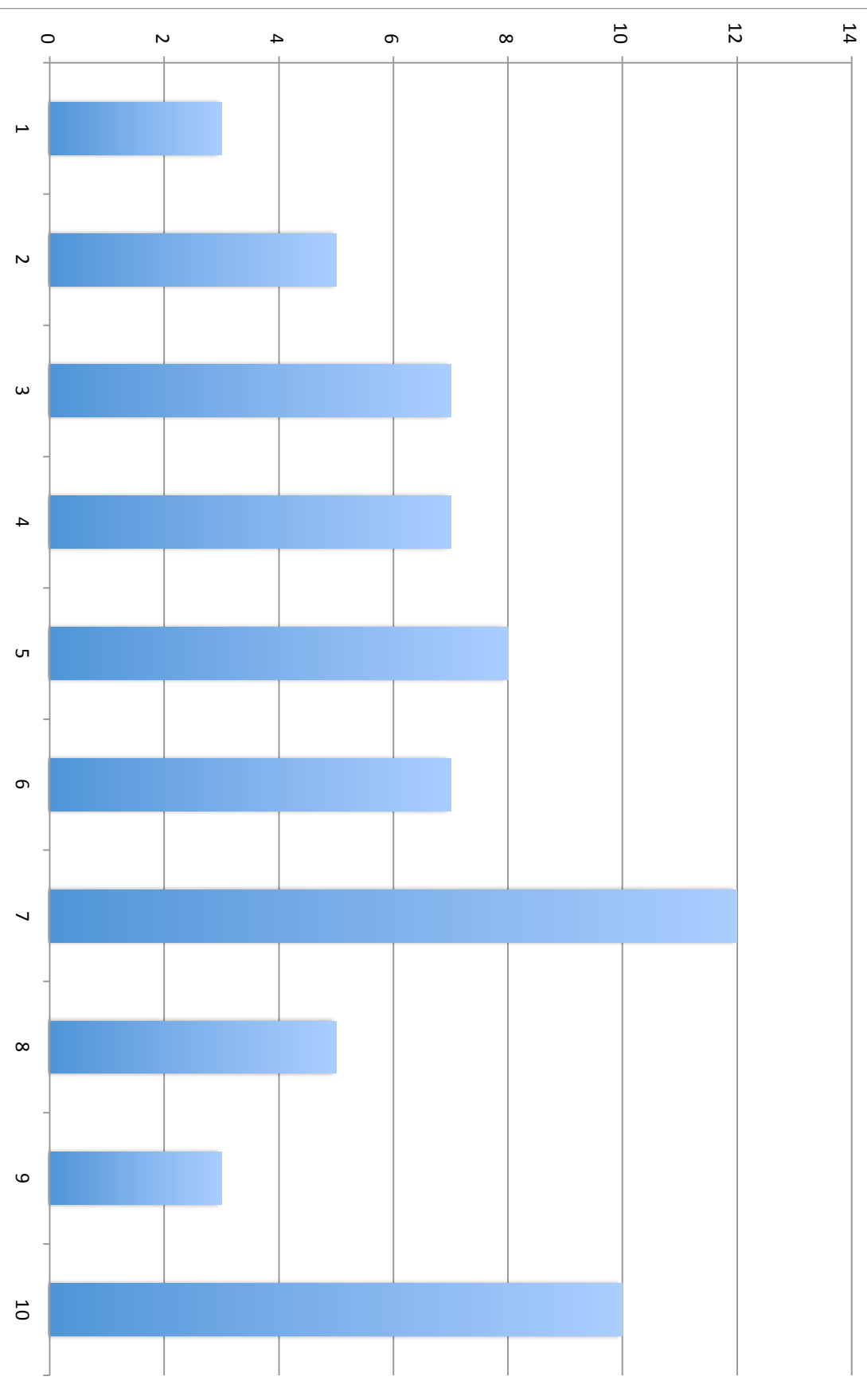


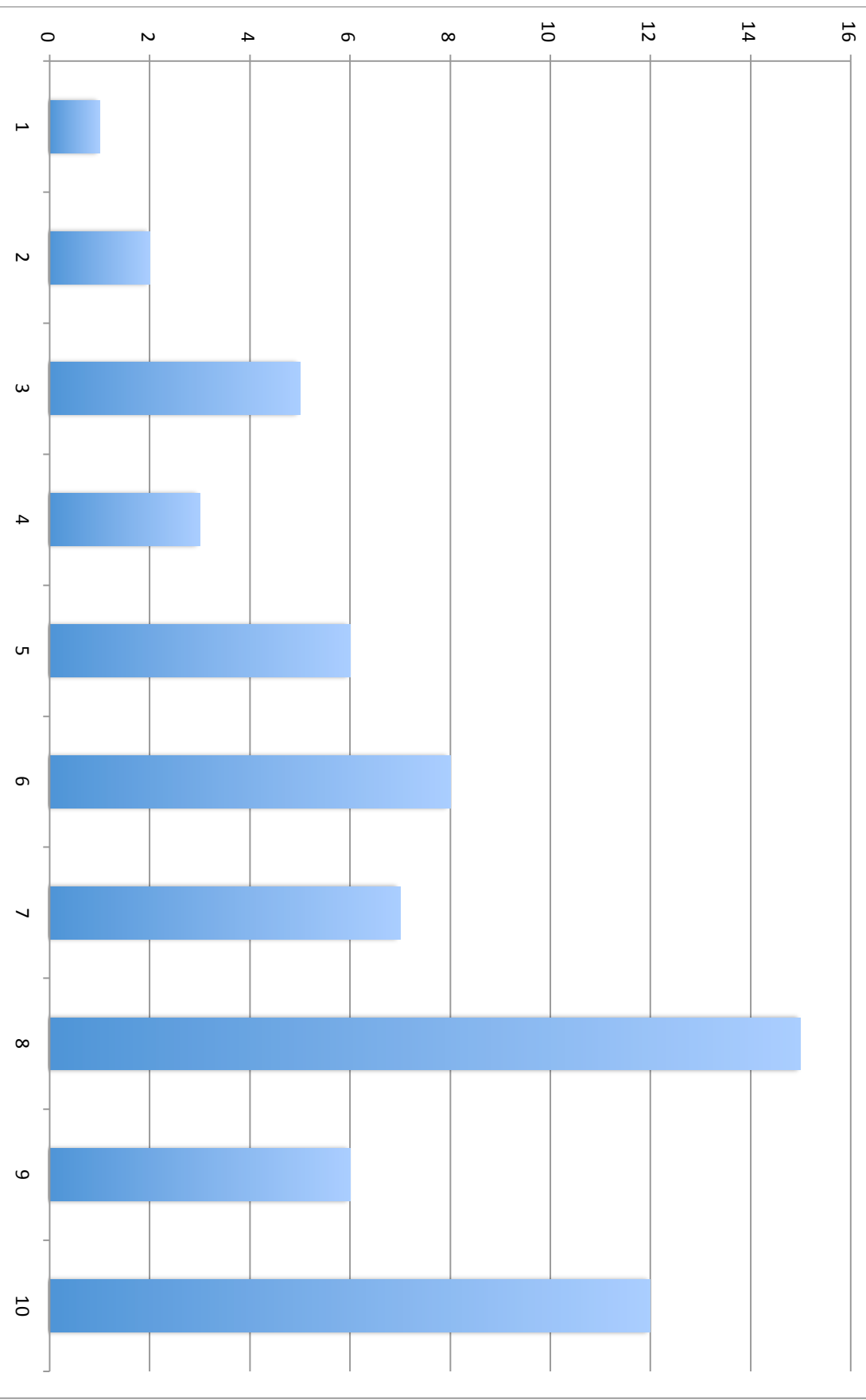
Interest in Chef



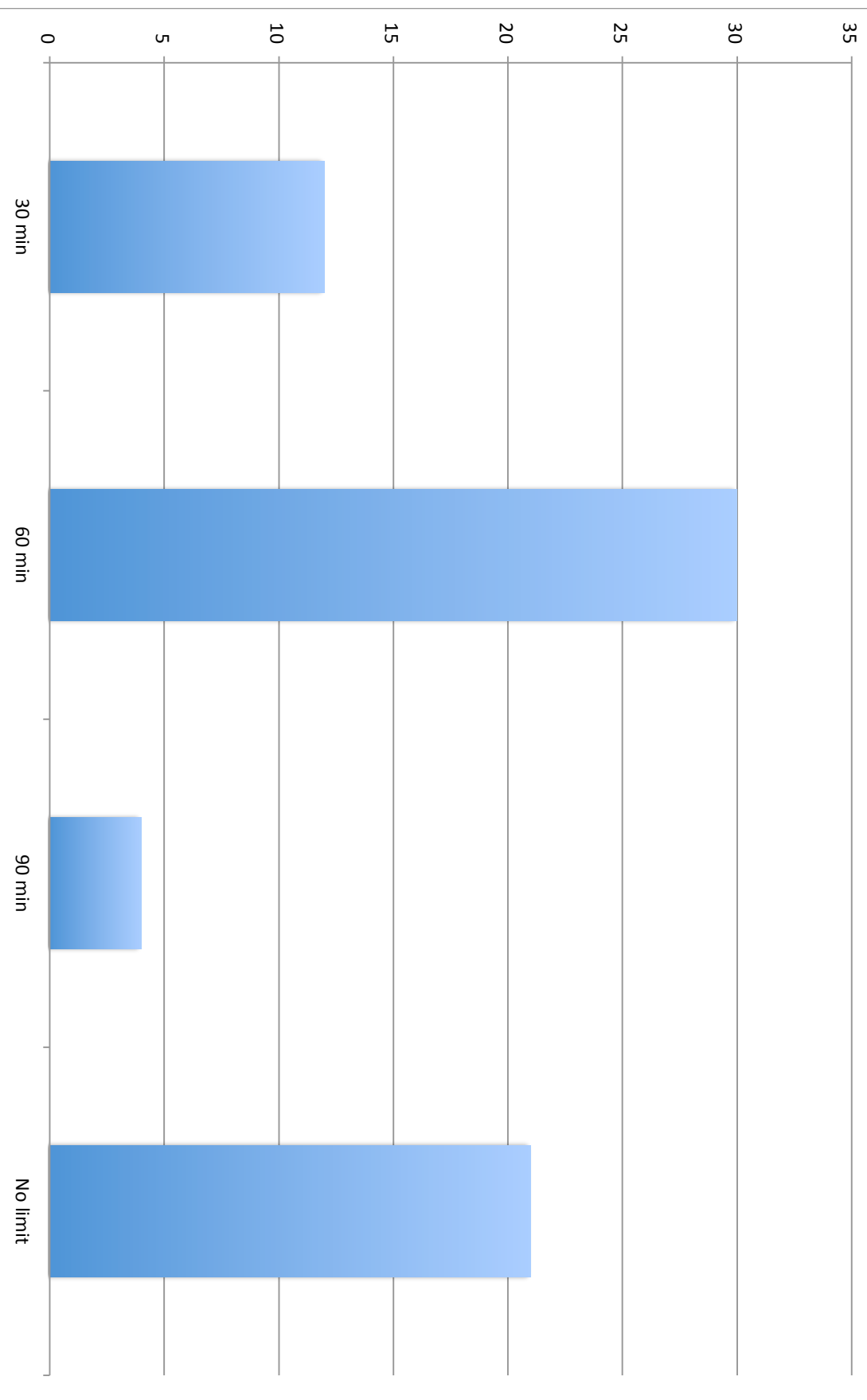
Interest in other CM



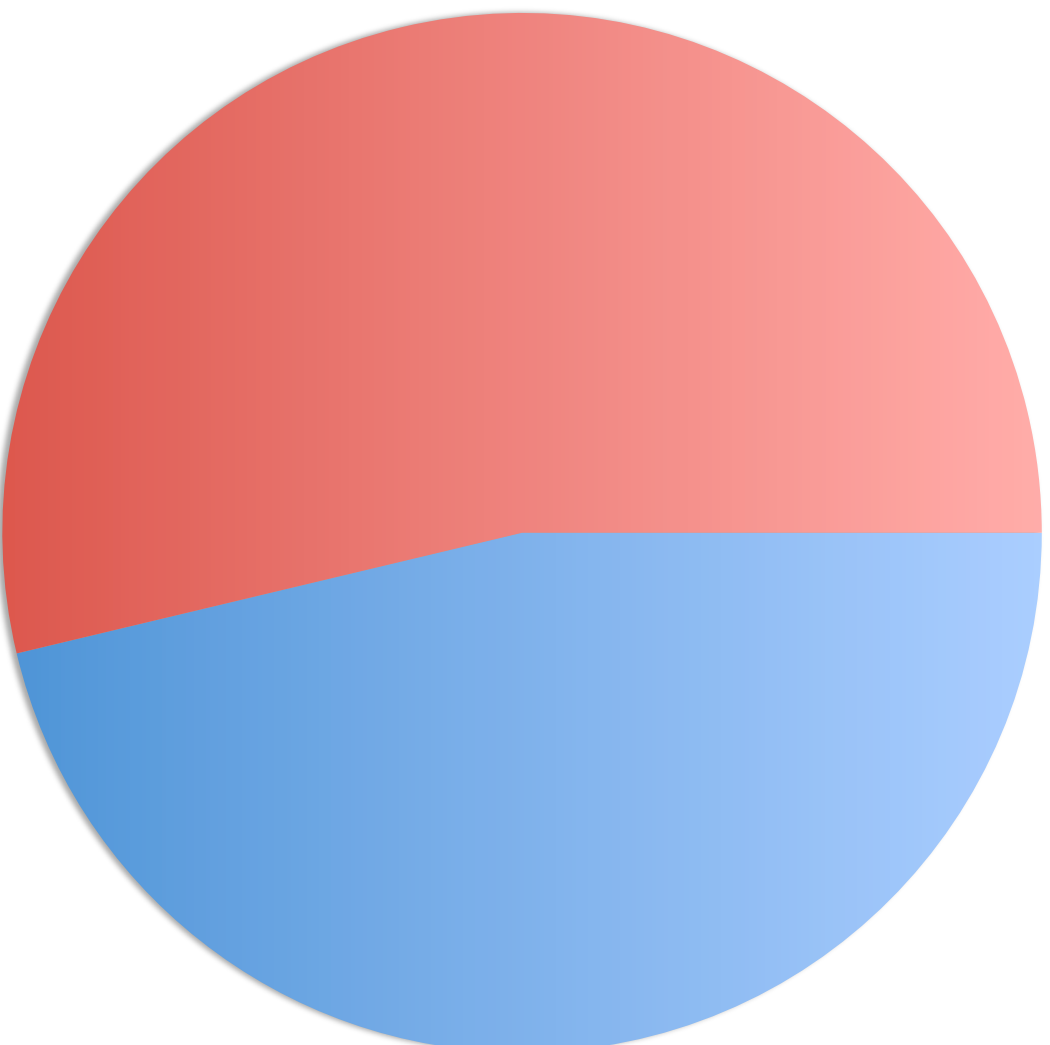
Usefulness of What's Cookin'



How long should the show be?

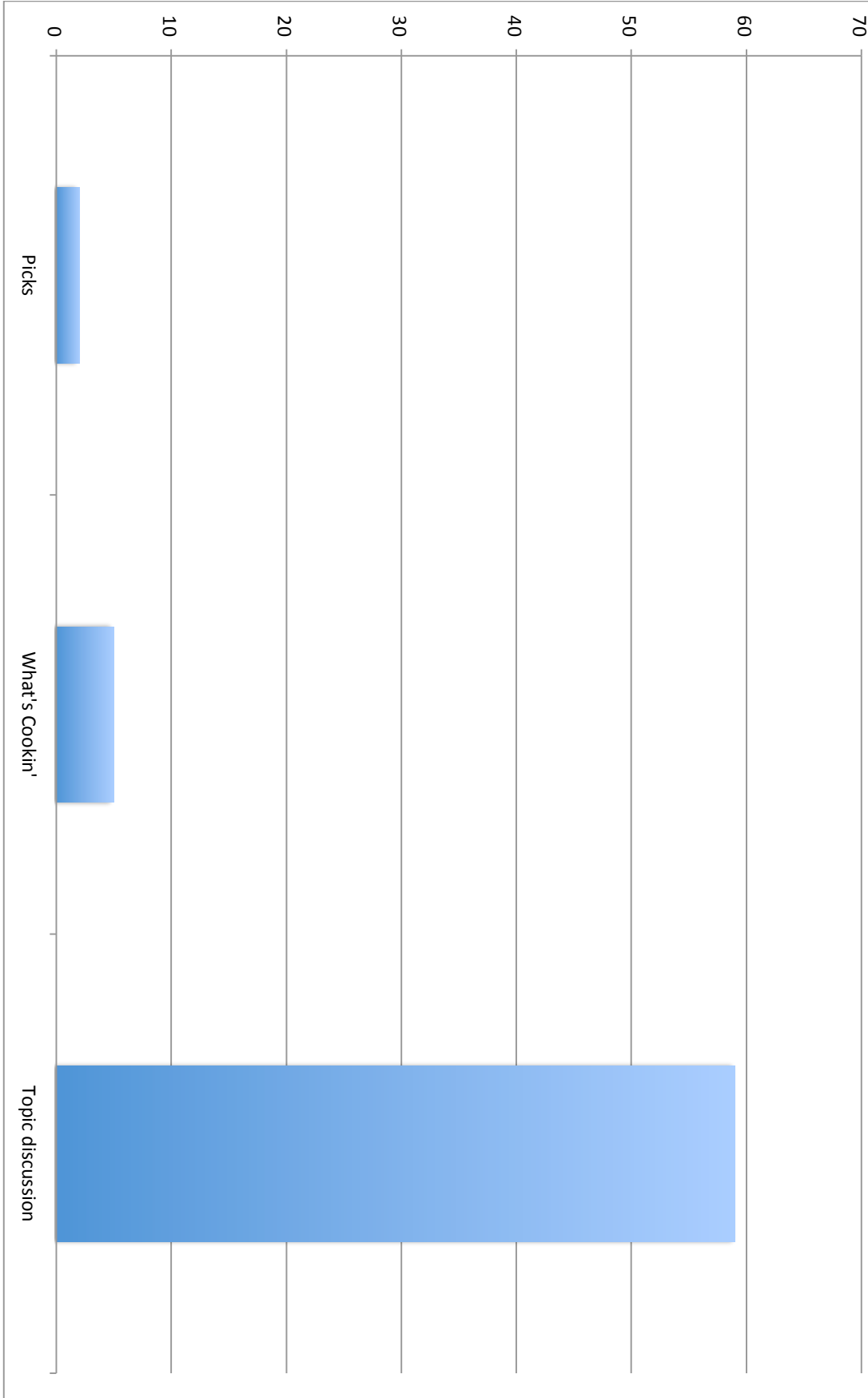


Separate What's Cookin' Podcast

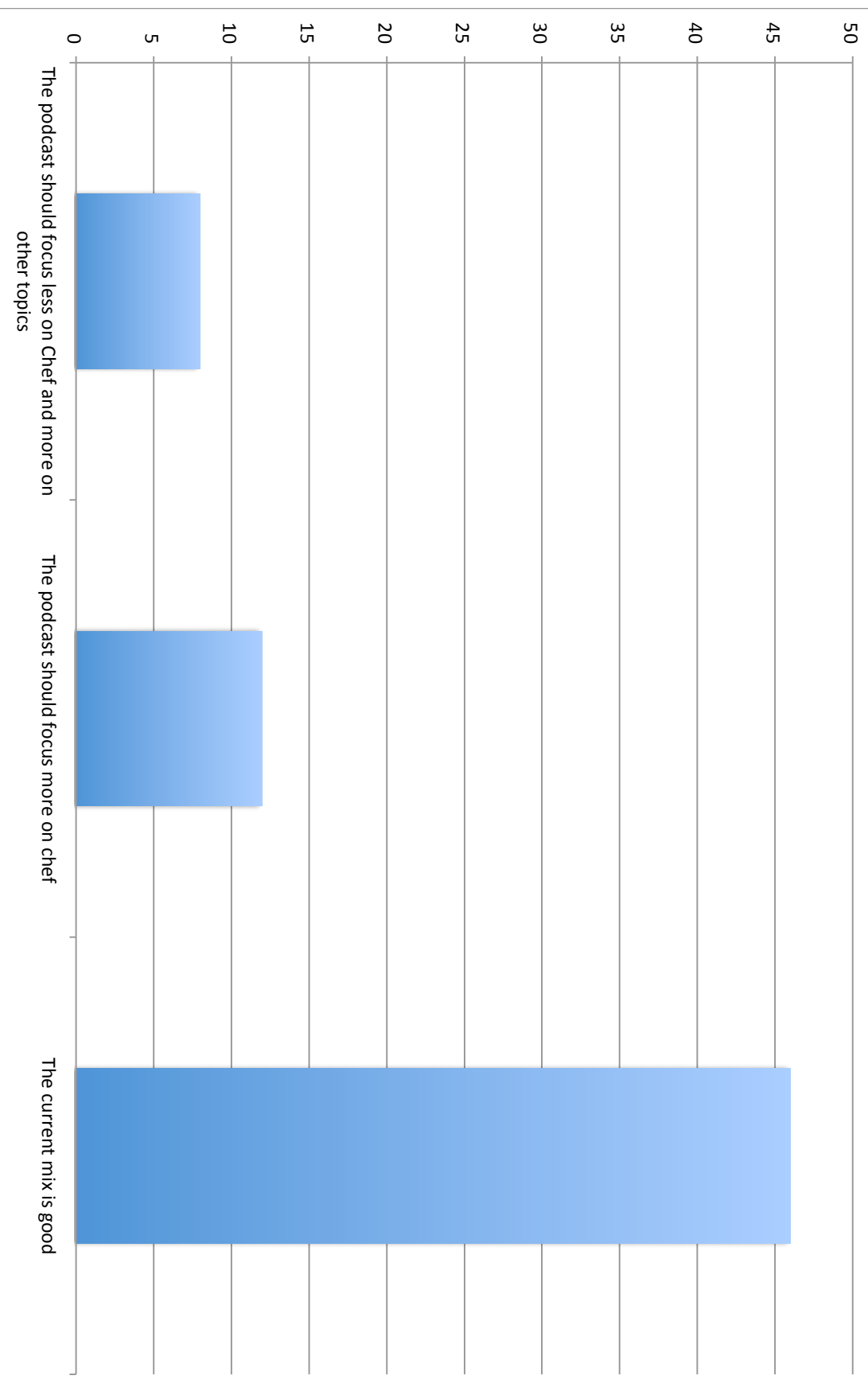


■ No
■ Yes

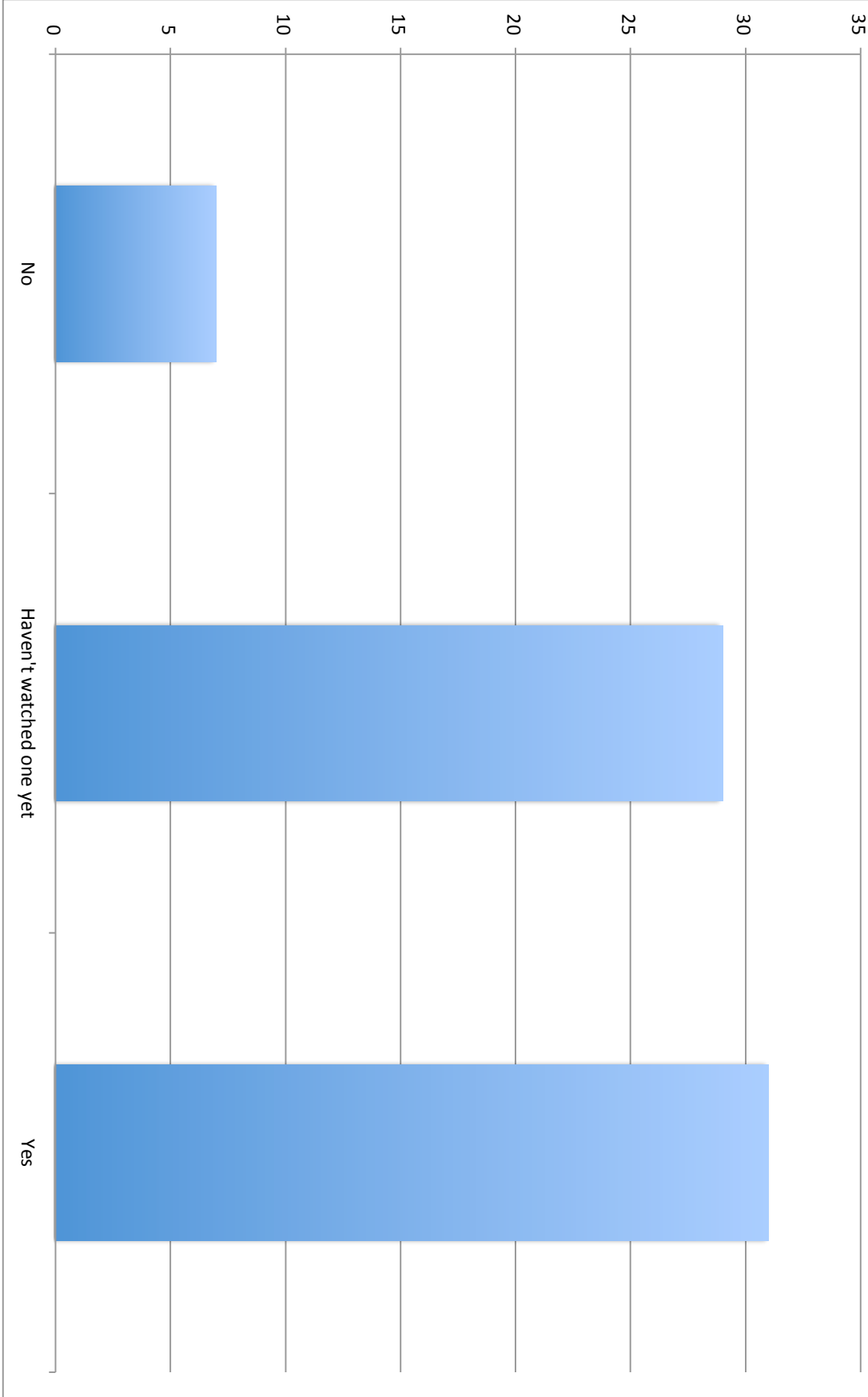
Favorite Segment



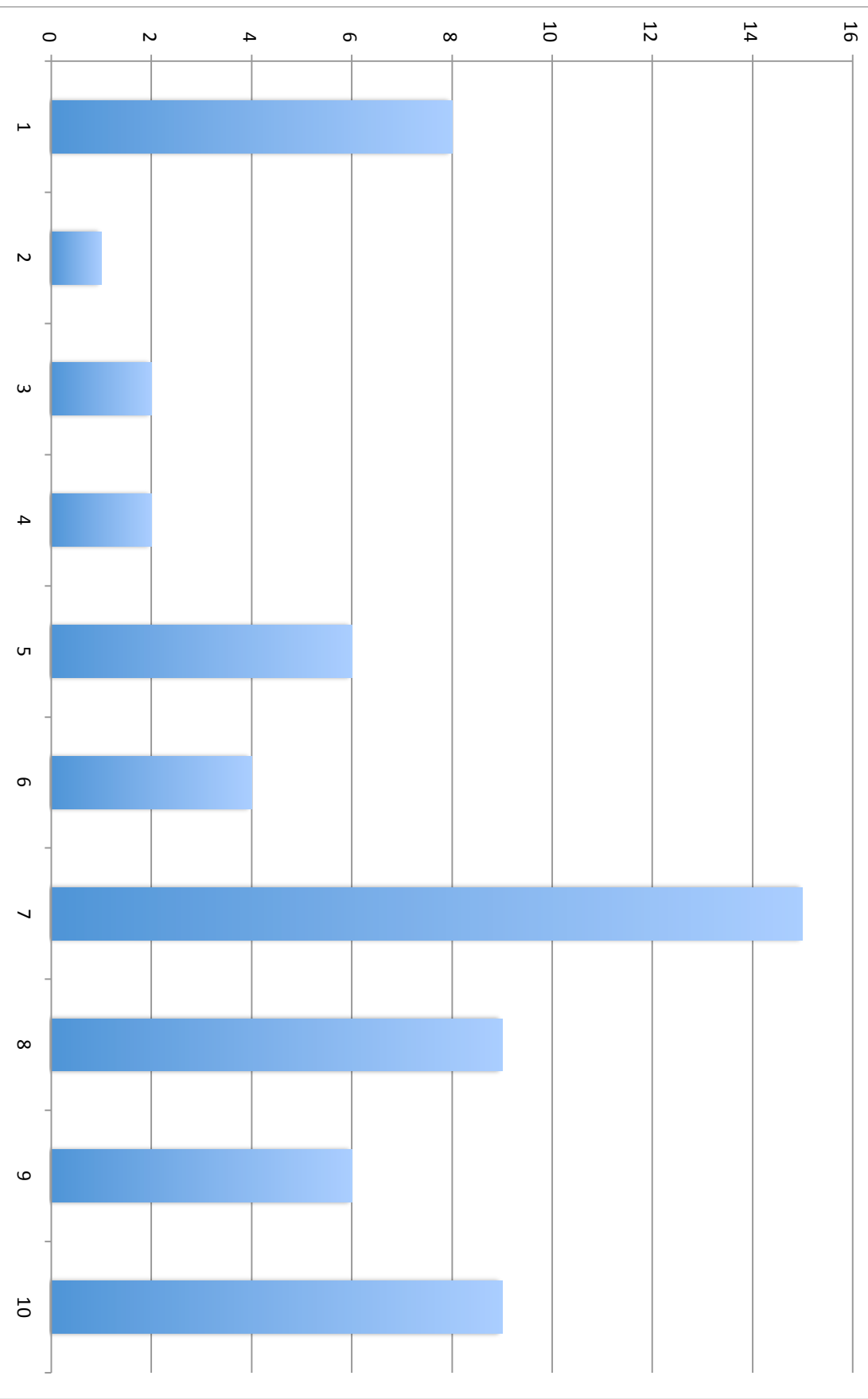
Focus of the Podcast



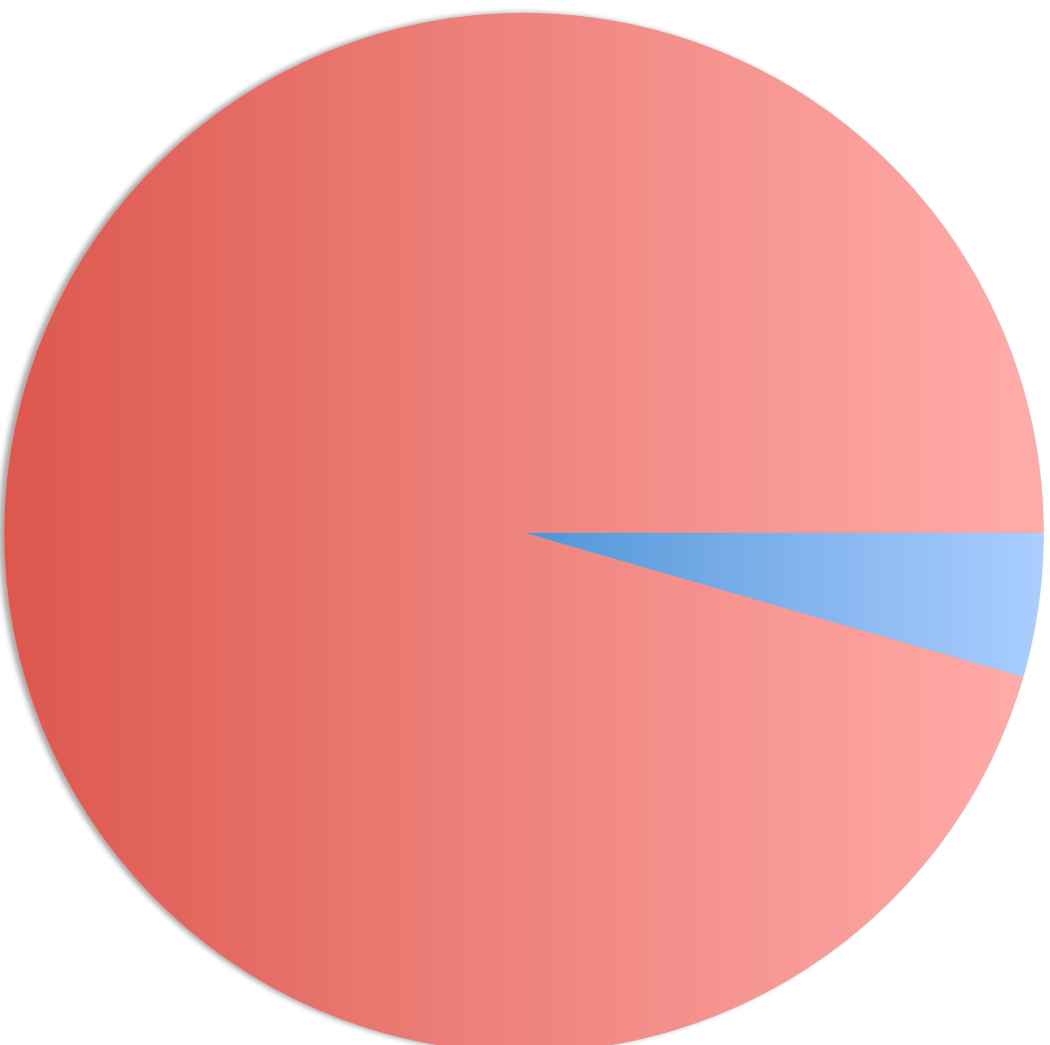
Are the Google+ Hangouts Useful?



Interested in watching a Hangout



Would you Subscribe to a What's Cookin' Email?



☐ No
☒ Yes

What topics would you like to see covered in greater detail?

- - How to migrate a Chef server to new hardware
- - How to contribute to cookbooks
- * Orchestration - how to do multi node releases. i.e. Something like chef-dominodes vs flock_of_chefs vs zookeeper
- * Internal vs external sources of configuration data - i.e. zookeeper vs data bags for coordination? LDAP vs data bags for configuration?
- * Configuring Java applications!
- * How ""enterprise"" shops use chef - particularly how it relates to heavy ITIL shops. Is Chef the cmdb? Is why run output sufficient to identify infra changes before CAB?
- * Continuous Delivery pipelines - how have they been setup with chef? Are stages defined by chef in code? In data bags? In separate system?
- * Orchestration tools
- * How people write and test LWRP
- * Anything on testing
- best practices for complex cases.
- best practices in chef development
- tools used with workflows
 - git
 - ruby
 - chef
- Case studies (with guests!)
- Cfengine
- Chef cookbook continuous integration/deployment.
- Chef is amazing tool, but what problems/difficulties do people still have (eg. workflow / cookbook version / etc)
- what other tools do people use with chef (jenkins, etc)
- Chef workflow
- Chef workflow in the real world with a team with mixed skills.

Continuous delivery practices with chef. There seem to be 10000 ways you *could* do things. So interested to hear how people actually do it. E.g. What Jenkins jobs do what and when.

The recent show that touched on this was awesome and I would love more.

- Cookbook development workflow;
How to better integrate tools like Berkshelf and Librarian into the workflow and with other tools such as Chefspec, Test-Kitchen and whatever else -- bugs and gotchas, what to use when and what not to use when;
How to lower the learning curve, or at least soften it a bit -- Chef's is too steep and things get out of control fast.

- Full stack management - end to end load balancer -> app server -> db server

Managing monitoring tools, dashboards, etc with Chef, using Search.

- I like hearing about Chef and do not want that to go away. Would also like to hear more about DevOps, Workflow and other tools.
- I would like to hear in more details about other operation tools. For instance, Rundeck, Gaphite and various web interfaces for it.
- I would like to see deployment tools covered in more detail, perhaps also deep dives on various tools in the ecosystem
- Integration with other tools. Managing different types of servers with chef (e.g. large clusters for one dedicated app vs. smaller isolated nodes).
- metrics scripting
- metrics, using chef in conjunction with web dev
- monitoring, cloud development and automation, scalability.
- More detailed info about getting started with cookbook testing and documentation.
- Orchestration
- Release engineering / code deployment all day every day! Cookbook writing
- best practices / common patterns
- security
- shuf
Chef in huge infrastructures (1 000 nodes and more)
- Workflow, and management issues
- Workflow, workflow, workflow. :-)
- Workflows
- Workflows that include chef and go all the way including orchastration....

Deep Deep Dive of a pattern all the way to code and why to use it. Some of the early history of Chef gave power to all ways.... but now patterns are emerging but why and which ones are the right one for a ""situation"" (a situation is also a pattern). What are the patterns and which of the chef practices and patterns are appropriate or better or optimal for which and why?

- Workflows, best practices, the big picture - we are sysadmins after all, we care more about the big picture than the details.

What's your favorite aspect of the show?

- content, informative
i always walk away with something learned
- Everything! Sorry if that is not helpful. I enjoy the show. Keep up the good work. I like the Google Hangouts as well.
- Everything? Great resource, look forward to every episode.
- Get an update from the community
- Good selection of guests
- Great guests and topics
- Great guests.
- hearing about best practices in this fast evolving space, and new tools to do that.
- Hearing best practices and case studies for how people are using technologies.
- Hearing from people who are already working the way I want to work with my team
- hearing how different companies/people use Chef
- How others are utilizing and expanding Chef.
- humour
depth
- I find the topic discussions to be the most useful. The guests you have on with specific experience have been invaluable.
- I love hearing about how other people have solved Ops problems whether that is with a cookbook or how they evaluate potential hires.
- Info about Chef. :)
- Insane relevance to what I need and want to accomplish at work.
- Personalities, enthusiasm, news
- The collected sysadmin knowledge mixed with the dev perspective.
- The conversational aspect. I prefer when multiple views are presented and discussed over the shows where a single person talks for an hour.
- The effort to provide an open community and discussion around chef and devops.
- The interesting guests make the show more interesting.
the interview

- The people you interview and their information.
- The podcast manages to get some really good guest speakers
- The quality in terms of general knowledge about the topics, and the informal tone.
- The topic discussion

What's your least favorite aspect of the show and why?

- Audio quality is lacking at times.
- Audio quality is often difficult to listen to.
- I don't care for the What's Cooking segment. Too much data about cookbooks I probably won't use.
- I love the show and will take all I can get. It's truly my favorite podcast by a long shot.
- I wouldn't say it's least favorite... but What's Cooking would be much more useful as an email blast. I won't say it's a least favorite because I've found out about a cookbook or project I was not aware of via What's Cooking. It'd be much easier for me personally to consume by email similar to the Devops Weekly newsletter.
- it's too long. Quite often, there are random
- most of the time I skip the what's cooking largely because with twitter there's rarely anything mentioned that i've not already read about new cookbooks / cookbook updates in details
- none really
- None.
- Not totally sure.
- nothing
- Nothing really
- Often poor sound quality, sometimes very difficult to listen to.
- picks are a bit hit and miss. Some of them are great, some less so.
- Some episodes seem to ramble without much structure
- Sometimes it seems to go too long :-)
- Sometimes the show seems a bit too scripted, like you guys are reading from a piece of paper. But when you get down to the chats, it gets very natural.

- Sometimes the topic discussion isn't very focussed, causing a long run time and ~20 minutes of off-topic discussion (fine when there's a point)
- sound isn't too good
- The introductions are sometimes confusing and disjointed. I'd like to have clearer guest introductions before jumping deep into the topic.
- The jingle is too loud, it sounds like those over-compressed TV ads, and the audio quality of the show is not the best -- not to mention too low at times. This is very annoying, because it makes me pump up the volume to listen to the show, but when the jingle kicks in it gets deafening at times.
- There's too much theory in some of the interviews. I love some thought provoking theory stuff, but I almost never have the time to follow up on it.
- Too much podcasts, too less time to listen. The frequency is kinda high/the show are often long. High frequency with shorter episodes or lower frequency with current length would be better.
- What's cookin' :) Sometimes it's a long string of things I'm not using or interested in!
- What's Cookin'. It just feels like it's being read from a script and would be better served as static (online or email) content.
- You guys have a great show and appreciate everything you do. But you must edit and curate your interviewees. You let people ramble on too long. You should interject and ensure the show is on track with whatever topic you are focused on.

What topics would you like to see covered in the Google+ Hangouts?

- Again, workflows and how to lower the learning curve. :)
- any related
- Anything.
- anything
- best practices with ruby, specifically how it is used in chef. Deep dives into the chef source.
- chef best practices.

We need good information on how to do development with chef. Many people coming to chef are from IT which has less experience with development methodologies and tools.

- deeper dive into the topic discussion of the week's show
- Discussion on chef testing.

- Discussion on keeping the chef server up-to-date with the latest cookbooks.
- Further discussion from recent episodes.
- Office-hours type Q/A.
- Hangouts seem less valuable to me, as they seem more a gathering of current influencers in the chef space, and topics seem aimed at that as an audience level rather than a chef user. I think discussions involving state of the
- idempotency and how resources achieve it deeper ruby in recipes
- Integrating RunDeck with Chef. Alternative UI's/ Dashboards for Chef.
- more workflow talk is good!!
- Overall development best practices / live cookbook reviews
- Recipe writing BP, screen demos of various tools (testkitchen, berkshelf) code and patterns.
- The current topics have been good. The same topics as mentioned in show ideas might be interesting as follow ups to shows.
- The google hangout should be more of a panelists taking "calls" from people who listened to the previous episode of FoodFight
- The hangouts are insanely useful to hear how people are actually solving real world issues. I'd keep focusing on specific aspects of operations & development overlap. It's also really interesting to watch how people's stories combine and begin to shape a consensus on community best practices.
- Definitely keep these up! The Hangout + Youtube archive is also very easy to use.
- Workflow, workflow, workflow
- Workflows!!!
- Would love to see someone take cookbook with no tests, write them, and test it all with test-kitchen