

<b>nathenharv...</b>	*** MEETING STARTS ***	
	<i>holoway does the magic fingers</i>	12:03 PM
	Agenda: <a href="https://github.com/chef/chef-community-irc-meet...">https://github.com/chef/chef-community-irc-meet...</a>	12:03 PM
<b>btm</b>	being able to scroll up a couple pages and see last weeks meeting is both convenient and disappointing that we don't have more conversation in here.	12:03 PM
<b>holoway</b>	okay, we're going to start with sdelano talking about the work we're doing with nightlies, feature flags, etc	12:03 PM
<b>sdelano</b>	ohai!	12:03 PM
<b>holoway</b>	then new PRs, then Old PRs,	12:04 PM
<b>sdelano</b>	Nightlies from Chef!	12:04 PM
	We're currently on a mission to make available nightly builds of all of the software that CHEF ships.	12:04 PM
<b>jonlives</b>	yay	12:04 PM
<b>sdelano</b>	The underlying desire of this mission is to get bug fixes and features in development into the hands of users and customers as quickly as possible.	12:04 PM
	Nightly builds, however, are just the start of this journey. The end goal is to have more frequent, even continuous, releases of CHEF software.	12:05 PM
	We've started by experimenting on the Chef Server, and if you've been paying close attention you might have noticed that we've been releasing this software about once every other week.	12:05 PM
	Previously, the Chef Server had a release cadence of roughly once every other quarter, and the massive risk involved in soaking that change has resulted in pain for both CHEF and our users and customers.	12:05 PM
	<i>jeffbyrn_ is now known as jeffbyrnes</i>	12:06 PM
	Along with these more frequent releases, we've changed a bit about the way that we're building the software. First and foremost we've been adding feature	12:06 PM

	flags and incrementally delivering new features.	
	The Key Rotation (incrementally delivered) and PolicyFile (feature flags) features are great examples of this. Shoutout to kallistec for getting the ball rolling on safely integrating PolicyFiles into the Chef Server. Awesome work so far!	12:06 PM
	The next step is getting these features into your hands faster than we have before.	12:07 PM
	In the next few weeks, we'll have the infrastructure in place to make nightlies available via our PackageCloud apt and yum repositories. We're also beginning work on rolling out another "channel" for OmniTruck, the web service that backs the `curl install.sh   bash` installation method for chef-client and chefdk. This will allow users outside of the apt and yum ecosystems to get in on the nightly action.	12:07 PM
	<i>chip-- is now known as chip-</i>	12:07 PM
btm	client moved to a new internal cluster ci/cd cluster recently which was building nightlies, but the functional tests aren't all passing on it yet so the git trigger and nightly trigger are disabled for a bit. when we turn them back on, you can get them from 'install.sh -n -p' until we simplify those flags.	12:08 PM
sdelano	In addition to the PackageCloud / OmnitTruck action, we're in the process of rebuilding the CI infra from ....	12:08 PM
	^^ what btm said	12:08 PM
holoway	was that a mic drop? :)	12:09 PM
sdelano	Once we've got the new chef-client chefdk CI infrastructure up, we'll take the lessons that we're learning on the server side and apply them to the client as well	12:09 PM
ranjib	very nice.	12:09 PM
jonlives	this is awesome stuff	12:09 PM
adamleff	This is sweet - thanks team!	12:09

		PM
thom	yep, totally great	12:10
		PM
ssd7	(for those who follow chef-server development, also note that we are re-arranging and merging underlying repos to make it easier to make changes)	12:10
		PM
sdelano	holoway: that's all. I fumbled the mic	12:10
		PM
holoway	for background, this is part of what we think can make the LT process more meaningful	12:11
		PM
	if we can get the release and quality process higher, it should be easier to have LTs and Maintainers make decisions that get immediately reflected in the source	12:11
		PM
	and then shipped	12:11
		PM
stevenmur...	oh yeah!	12:11
		PM
holoway	which helps remove one of the main "chef software" road-blocks today, which is that only we can release	12:11
		PM
jonlives	good call.	12:12
		PM
jtimberman	one thing we're also working on is getting metrics about those changes, how long it takes for a PR to go from "opened" to "merged" and also "in the package you can download"	12:12
		PM
miketheman	will the build process cluster be visible to non-Chef Software employees?	12:12
		PM
	<i>jtimberman literally writing some pseudocode for that right now.</i>	12:12
		PM
holoway	sdelano: thank you for all the work, and the update - long live the legion of builds	12:12
		PM
jonlives	amen to that	12:13
		PM
holoway	miketheman: I think we need to work in that direction	12:13
		PM

<b>ssd7</b>	miketheman: That is on the list of "stuff to figure out" but we've got a lot of more basic problems in front of it first	12:13 PM
	oh, what holoway said :)	12:13 PM
<b>holoway</b>	miketheman: we prioritized just getting it to work all the way through first	12:13 PM
<b>sdelano</b>	miketheman: not currently, but we're moving that direction. lots of things are moving around right now and when they settle, we'll be in a better place to make them accessible	12:13 PM
<b>holoway</b>	miketheman: then we can talk about how to get a read-only view	12:13 PM
<b>miketheman</b>	holoway, ssd7: makes sense. I really liked when homebrew shows you how your build/Pull Request is doing on jenkins, a la travis	12:14 PM
<b>holoway</b>	ya	12:14 PM
<b>miketheman</b>	awesome news, thanks!	12:14 PM
<b>holoway</b>	ok - on to new bidness	12:14 PM
	(I think it's nice to flip-flop new business/old business on the schedule, btw)	12:14 PM
	<a href="https://github.com/chef/chef-rfc/pull/93">https://github.com/chef/chef-rfc/pull/93</a>	12:14 PM
<b>sdelano</b>	miketheman: I'll go check that travis -> jenkins action out. thanks for the suggestion	12:14 PM
<b>holoway</b>	on demand cookbook libraries	12:14 PM
	miketheman: yeah, having stage 0 be our invisible jenkins cluster putting comments on the PR is a great idea	12:15 PM
	mr jkeiser around?	12:15 PM
<b>sdelano</b>	holoway: vacation iirc	12:16 PM

adamedx	hes on vacation	12:16
		PM
ranjib	we should just merge this RFC, i think we have consensus here	12:16
		PM
thom	agreed	12:16
		PM
holoway	ah	12:16
		PM
miketheman	question	12:17
		PM
ranjib	jonlives, you have any more comments on this?	12:17
		PM
miketheman	this adds a new thing to metadata.rb that affects how chef-client behaves	12:17
		PM
jonlives	ranjib: I'm good on that one	12:17
		PM
miketheman	we've seen many other items be added to metadata - supports, suggessts, attributes, etc - that don't actually affect how the client behaves	12:17
		PM
ranjib	those are for documentation purpose	12:18
		PM
holoway	(well, that's because I'm not a very good developer sometimes, and hadn't learned YAGNI yet- so I just took the most full featured spec available, which is debian packaging, and cargo culted the fiedls)	12:18
		PM
miketheman	at what point do we start to consider "what is actually relevant" in metadata.rb?	12:18
		PM
jtimberman	and then there's metadata things that affect other tools like stove and berks/ridley	12:18
		PM
holoway	but I agree	12:18
		PM
	we should merge this	12:18
		PM
	anyone against?	12:18
		PM

jtimberman	holoway: i think we should note that this would affect external tooling	12:19
		PM
miketheman	esp. if other tools `require 'chef'`	12:19
		PM
	and then read cookbooks	12:19
		PM
	to load libraries	12:19
		PM
kallistec	miketheman: I'd be in favor of removing the most egregious stuff (supports/suggests) or modifying it to be clear that it's only documentation	12:20
		PM
jtimberman	when we added issues_url and source_url to metadata for supermarket to consume, those tools had issues when evaluating metadata.rb (to upload, in stove) or to consume (ridley for `berks`)	12:20
		PM
jonlives	we're gonna have this again with chef_version and ohai_version too	12:20
		PM
kallistec	miketheman: but that's off topic for right this second	12:20
		PM
holoway	because they wrote their own custom classtes to parse metadata.rb?	12:20
		PM
miketheman	kallistec: agreed - that's a bigger problem for sure	12:20
		PM
jtimberman	jonlives: yup	12:20
		PM
	holoway: right	12:20
		PM
	they validate which metadata fields are allowed.	12:20
		PM
holoway	well, I got 99 problems but their custom code ain't one - but we should reach out to the authors	12:20
		PM
	please comment on that PR, and lets reach out	12:21
		PM
	(hey, jonlives - we're about to break stove)	12:21
		PM

	(twice!)	12:21
		PM
jonlives	holoway: I didn't write that one ;)	12:21
		PM
jtimberman	we ship at least ridley in chefdk	12:21
		PM
holoway	aww	12:21
		PM
	I was going to be awesome	12:21
		PM
jonlives	:D	12:21
		PM
martinisoft	That was sethvargo	12:21
		PM
someara	that's a seth vargo joint	12:21
		PM
holoway	my point stands - lets fix those tools	12:21
		PM
jonlives	agreed	12:21
		PM
holoway	add a list to the PR	12:21
		PM
	still the right thing to do	12:21
		PM
jtimberman	i will do that.	12:21
		PM
someara	this gives me an idea for another RFC!	12:21
		PM
holoway	ok - next topic	12:21
		PM
miketheman	those tools should likely use a common metadata parser to work, instead of each tool reinventing the wheel	12:22
		PM

<b>jonlives</b>	and a community summit session at chefconf :p	12:22	PM
<b>miketheman</b>	anyhow	12:22	PM
<b>holoway</b>	this feels like a no-brainer	12:22	PM
	<a href="https://github.com/chef/chef-rfc/pull/95/files">https://github.com/chef/chef-rfc/pull/95/files</a>	12:22	PM
<b>github-bawt</b>	[chef-rfc] ranjib closed pull request #86: Let chef-client consume github release tarballs in local mode (master...github_release) <a href="http://git.io/F4dk">http://git.io/F4dk</a>	12:22	PM
<b>thom</b>	someara: i think jon postel beat you to that one	12:22	PM
<b>holoway</b>	specify the quantity of each role in the maint process	12:22	PM
	tl;dr, there is one project lead, one LT per area, and many maintainers	12:23	PM
<b>jonlives</b>	+1 to merging	12:23	PM
<b>kallistec</b>	+1	12:23	PM
<b>holoway</b>	anyone against?	12:23	PM
	30 seconds	12:23	PM
<b>github-bawt</b>	[chef-rfc] ranjib closed pull request #82: Allow specifying partial templates during bootstrap (master...bootstrap_partial) <a href="http://git.io/pX1p">http://git.io/pX1p</a>	12:23	PM
<b>miketheman</b>	only one LT? how is that military-related?	12:23	PM
	:)	12:23	PM
<b>thom</b>	can we fix the typo i just noticed? :)	12:23	PM



	but yes, do it	12:24
		PM
holoway	yes!	12:24
		PM
btm	thom: you can fix one typo, but not mulitple ones.	12:24
		PM
thom	:D	12:24
		PM
jonlives	thom: sure, submit an RFC for a typo-fixing process ;)	12:24
		PM
cmluciano	zing	12:24
		PM
	<i>holoway dies</i>	12:24
		PM
miketheman	also, the use of `role` here is mildly confusing with Chef's `role`	12:24
		PM
martinisoft	+1 to merge, after the typo fix :D	12:24
		PM
holoway	<a href="https://github.com/chef/chef-rfc/pull/96">https://github.com/chef/chef-rfc/pull/96</a>	12:24
		PM
	restricting search to the current environment by default	12:24
		PM
jonlives	miketheman: that's just how we role.	12:24
		PM
	<i>martinisoft makes new RFC to rename LT to 'Highlander'</i>	12:24
		PM
someara	-1 backwards compat nightmare	12:25
		PM
ranjib	-1 on this	12:25
		PM
jtimberman	jonlives: *groan*	12:25
		PM

jonlives	I am very very -1 on #96 being enabled by default	12:25
		PM
ranjib	the meaning of environment is very different from org to org	12:25
		PM
	<i>miketheman slaps jonlives around a bit with a large trout</i>	12:25
		PM
thom	yeah, i am -1 on my own RFC at this point	12:25
		PM
ssd7	As an alternative, what if we make -E work with search as an option?	12:25
		PM
martinisoft	-1 on this as well. It's not a fair assumption (especially with Solo/Zero users)	12:26
		PM
adamedx	surprised -E doesn't already work	12:26
		PM
jtimberman	I'm -1, "my_query:* AND chef_environment:#{node.chef_environment}" is fine. It's a thing we've done for years w/o issue IMO.	12:26
		PM
adamedx	I remember seeing a bug on that at least	12:26
		PM
ranjib	ssd7, that will involve outof band query modification?	12:26
		PM
holoway	I'm +1 on adding an option to search to make typos less relevant	12:26
		PM
miketheman	we also use the search pattern defined all the time, I do wish there was a "easier" method	12:26
		PM
holoway	we have a few other things - partial search being default, safe search	12:26
		PM
thom	as lamont points out, ACLs might be a correct answer to some of this	12:26
		PM
github-bawt	[chef-rfc] btm pushed 1 new commit to master: <a href="http://git.io/pXDa">http://git.io/pXDa</a>	12:26
		PM
	chef-rfc/master f302dd3 Bryan McLellan: Specify the quantity of each role	12:26
		PM

<b>miketheman</b>	-1	12:26
		PM
<b>thom</b>	jtimberman: normally you like being lazy :P	12:27
		PM
<b>ranjib</b>	we should offer a `query` DSL. backed by partial search for that kinda stuff. keeping search as it is	12:27
		PM
<b>ssd7</b>	ranjib: I was thinking something like thom's recent addition to the knife-status command	12:27
		PM
<b>jonlives</b>	doing it in ACLs feels like a nicer option to me	12:27
		PM
<b>holoway</b>	so I'm -1 on this RFC, but I'm +1 on a search reform RFC that keeps back-compat on the current interface, but rolls up a bunch of great changes	12:27
		PM
<b>jtimberman</b>	thom: that means i can copy and paste that code everywhere though	12:27
		PM
<b>github-bawt</b>	[chef-rfc] btm closed pull request #95: Specify the quantity of each role (master...role-sizes) <a href="http://git.io/x29r">http://git.io/x29r</a>	12:27
		PM
<b>holoway</b>	ranjib: BINGO	12:27
		PM
	^^ what ranjib said	12:27
		PM
<b>thom</b>	ranjib/holoway: yup. i'll take that and see what i can do	12:27
		PM
<b>jtimberman</b>	ranjib: great idea! I look forward to your RFC :D	12:27
		PM
<b>holoway</b>	(also awesome - if that query API included multiple backends)	12:27
		PM
<b>ranjib</b>	ssd7, yeah, but search is a lot low level API.	12:28
		PM
	exactly, have pluggable backend. with default to chef search	12:28
		PM
<b>jonlives</b>	that sounds delicious.	12:28
		PM

<b>ranjib</b>	we can have etcd, zk for strong consistency needs, hlld for less exact needs etc	12:28 PM
<b>thom</b>	it sounds delightful!	12:28 PM
<b>adamedx</b>	do it	12:29 PM
<b>jonlives</b>	somebody make that ranjib an awesome comm....oh no wait, we already did.	12:29 PM
<b>holoway</b>	next	12:29 PM
<b>Boringgeek</b>	#golfClap	12:29 PM
<b>someara</b>	eh... why not zk_search hlld_search?	12:29 PM
<b>github-bawt</b>	[chef-rfc] adamhjk closed pull request #96: Restrict search to current environment by default (master...tm/restricted-search) <a href="http://git.io/xMle">http://git.io/xMle</a>	12:29 PM
<b>holoway</b>	<a href="https://github.com/chef/chef-rfc/pull/97">https://github.com/chef/chef-rfc/pull/97</a>	12:29 PM
	remove roadmapping as an LT responsibility	12:30 PM
<b>miketheman</b>	MERGE CONFLICT	12:30 PM
	whoosp	12:30 PM
<b>holoway</b>	hehe	12:30 PM
	I want to re-iterate what is in the comments	12:30 PM
	I want the LTs involved in the future direction (and the maintainers, and the rest of the community) of Chef	12:31 PM
	it's unclear to me how to run a 'process' that results in a traditional software roadmap - ie, a grid divided into quarters and a row per area	12:31 PM

	and for sure, I don't think a structure that has LTs telling people what to do is actually the right way to think about running an open source project	12:32 PM
jonlives	agreed	12:32 PM
chip-	I think that traditional roadmap gets less relevant the more Chef ships features incrementally / with flags	12:32 PM
holoway	cause, we all volunteer (even the chef employees - in general, we don't make anyone participate in the OSS process, unless its the only way to accomplish their goals)	12:32 PM
	and so - as meta context for the merging of this RFC	12:33 PM
miketheman	I think having an idea of what a component needs work on is helpful to other contributors - "feature X is something we don't really want right now for ABC reasons, feature Y is something cool and shiny"	12:33 PM
	so _some sort_ of public direction should be provided	12:33 PM
holoway	I say we run a session at the community summit pre-chefconf to come up with a simple place we can communicate what futures we understand	12:33 PM
	perhaps a ROADMAP.md	12:33 PM
jonlives	sounds good to me	12:33 PM
holoway	that LTs can just merge to at will	12:33 PM
	as one possible example	12:33 PM
miketheman	holoway: ding ding ding	12:33 PM
thom	miketheman: we do that somewhat with milestones for issues, too	12:33 PM
holoway	and we can see if it becomes a mess	12:33 PM
btm	let's just ship that. :)	12:33

		PM
<b>miketheman</b>	thom: agreed!	12:34
		PM
<b>holoway</b>	or, you know	12:34
		PM
	I would accept the Roadmap.md RFC	12:34
		PM
	:)	12:34
		PM
<b>jonlives</b>	that also means LTs can focus on the aspects of their components most relevant to them	12:34
		PM
<b>jtimberman</b>	yeah. using the issue tracker to ... track issues that need to be addressed, seems plausible ;).	12:34
		PM
<b>miketheman</b>	I think the point I'm saying is "I don't want to tell you how to produce a visible roadmap - but you should try to do it when possible"	12:34
		PM
<b>thom</b>	holoway: i'm not sure when would be the right time to write the roadmap rfc ...	12:34
		PM
<b>btm</b>	I'll work on a ROADMAP.md RFC, but I think we should merge 97 still.	12:35
		PM
<b>holoway</b>	me too	12:35
		PM
<b>jonlives</b>	I'm saying "I don't want to have to produce a roadmap, but still be able to say 'I really care about this'" :p	12:35
		PM
	yeah +1 on merging 97	12:35
		PM
<b>miketheman</b>	+1 on 96 merge	12:35
		PM
<b>holoway</b>	jonlives: I think the ROADMAP.md might do that	12:35
		PM
	just stick it in there	12:35
		PM
	:)	12:35
		PM

<b>miketheman</b>	97 that is	12:35
		PM
<b>thom</b>	yup. let's merge it	12:35
		PM
<b>jonlives</b>	holoway: right, I'm agreeing with you :p	12:35
		PM
<b>holoway</b>	jonlives: oh great!	12:35
		PM
	:)	12:35
		PM
<b>miketheman</b>	stop arguing about agreeing! :)	12:35
		PM
<b>holoway</b>	<a href="https://github.com/chef/chef-rfc/pull/100">https://github.com/chef/chef-rfc/pull/100</a>	12:35
		PM
	NO YOU STOP	12:35
		PM
	<i>holoway flips the table</i>	12:36
		PM
<b>miketheman</b>	LOL	12:36
		PM
<b>chip-</b>	Vehement agreement!	12:36
		PM
<b>holoway</b>	node state separation, - <a href="https://github.com/chef/chef-rfc/pull/100">https://github.com/chef/chef-rfc/pull/100</a>	12:36
		PM
<b>kallistec</b>	btm & holoway can we just use generic language like "provides guidance on future direction for their subsystem" ?	12:36
		PM
	back on #97 sorry	12:36
		PM
<b>btm</b>	kallistec: I'm happy to make words like that in a ROADMAP.md rfc	12:37
		PM
<b>holoway</b>	lets make words like that in the ROADMAP rfc	12:38
		PM

	ok, so for node state separation	12:38
		PM
	we have to do this	12:38
		PM
thom	it feels like 100 needs more fleshing out	12:38
		PM
holoway	but this isn't quite ready to merge	12:38
		PM
lamont_oc	well do we want to do the separate ACL'ing or not?	12:38
		PM
holoway	I feel like this was my largest mistake	12:38
		PM
btm	kallistec: and I promise to work on it within 24 hours.	12:38
		PM
kallistec	btm: ok	12:38
		PM
holoway	we need to separate desired from current state	12:38
		PM
thom	holoway: openid is swept under the rug now?	12:38
		PM
holoway	I think we need to address acls	12:39
		PM
	thom: we fixed that mistake!	12:39
		PM
lamont_oc	i could use some more feedback on if desired and current are separate ACLs	12:39
		PM
	<i>jtimberman chuckles</i>	12:39
		PM
holoway	(and I was ahead of mine time - the whole world uses token auth now, so fuck the haters)	12:39
		PM
thom	we really need to adress ACLs here	12:39
		PM



	lamont_oc: i think they aare	12:39
		PM
holoway	(I was a genius)	12:39
		PM
	(take that)	12:39
		PM
someara	how about we just make node attributes immutable for the duration of the chef-client run and call it a day	12:39
		PM
jonlives	lamont_oc: one question on that one - it says node.save saves to the runtime data...I don't see anything saying when then configuration data will be updated, if at all.	12:39
		PM
holoway	someara: that is not a solution	12:39
		PM
	someara: it's fraught with edge cases	12:39
		PM
chip-	someara: doesn't fly when you've got 125 CCRs/sec reporting in	12:40
		PM
lamont_oc	node.save would update the configuration data if it was dirtied	12:40
		PM
thom	doesn't immutability solve everything though?	12:40
		PM
holoway	putting a big lock around it just leads to the problems that locks have in distributed systems	12:40
		PM
miketheman	what if the run never completes, and the node is in immutable state forever?	12:40
		PM
holoway	ask cheeseplus how he feels about that	12:40
		PM
jonlives	lamont_oc: ah - I think that needs clarified a little, the languages used in the node save section was a little unclear on that.	12:40
		PM
chip-	I feel strongly that these should be separate ACLs. It's a very common request from THE ENTERPRISE that run_lists and environments be immutable.	12:40
		PM
holoway	so - I think thom is right, the ACLs have to be separate	12:41

		PM
btm	holoway: +10	12:41
		PM
jtimberman	chip-: +LONG_INT	12:41
		PM
thom	chip-: yar	12:41
		PM
	<i>someara retracts his comment.. E_DISTRIBUTED_SYTEMS_ARE_HARD</i>	12:41
		PM
holoway	with the 12 implementation defaulting to open	12:41
		PM
	and 13 defaulting to closed	12:41
		PM
lamont_oc	hmmmm, okay maybe when i changed i from explicitly PUT'ing to /nodes/:name/configuration and /runtime to just using /nodes/:name that became less obvious what i was doing...	12:41
		PM
kallistec	I think in that case though you need to make the configuration part more granular	12:41
		PM
miketheman	lamont_oc: node.save in a recipe, or node.save at the end of a run? Won't some attributes always be dirtied?	12:41
		PM
jonlives	lamont_oc: I understand "If it is detected that the configuration data has been dirtied then node.save will instead PUT to /nodes/:name. " once you explained it, but yeah I think it needs clarifying a little in the RFC	12:42
		PM
holoway	I think part of ti is /configuration and /runtime isn't quite right languatge	12:42
		PM
	it's really /desired and /current	12:42
		PM
kallistec	like tags and normal attributes you probably need to be able to modify	12:42
		PM
lamont_oc	the runtime attributes are always dirtied and always written, the configuration attributes usually aren't	12:42
		PM
jonlives	could we use "desired" and "current"?	12:42
		PM

<b>holoway</b>	to put it another way - you want to let the 'current' state be mutable and is often dirtied	12:42 PM
<b>jonlives</b>	oh holoway just said that. +1 to holoway's suggestion :p	12:42 PM
<b>holoway</b>	the 'desired' state is often not mutable by the node	12:42 PM
<b>chip-</b>	This also opens the gateway to having node objects in the repo, and modifying/uploading those as files without worrying about wiping current state, which would be :D	12:42 PM
<b>lamont_oc</b>	yeah i copied that from john k's original gist, i can use desired and current instead	12:42 PM
<b>ssd7</b>	I'd like to see the client-side accessors for this data split spelled out as well	12:42 PM
<b>miketheman</b>	holoway: disagree - nodes may make changes to their own run lists, changing the desired state for next run	12:43 PM
<b>thom</b>	+1 to desired and current	12:43 PM
<b>martinisoft</b>	Ooo, I like that chip-	12:43 PM
<b>holoway</b>	miketheman: which is fine, they can do that	12:43 PM
<b>miketheman</b>	(and such)	12:43 PM
<b>holoway</b>	you have ACLs on Desired and Current separately	12:43 PM
	implementation 1 is back-compat, with the default acl on desired being open	12:43 PM
<b>martinisoft</b>	I like desired and current as well	12:43 PM
<b>holoway</b>	but you move to desired being closed to the node itself	12:44 PM
	by default	12:44 PM

thom	yes	12:44
		PM
jonlives	I think given this is a pretty large change, expanding the rationale and motivation sections would be a good idea	12:44
		PM
holoway	it's mostly a plumbing change - I bet you can land the entire feature without breaking back-compat	12:44
		PM
	jonlives: yes	12:44
		PM
someara	I've never understood how a node attribute can simultaneously be input and output of a chef-client run	12:44
		PM
jonlives	ie, for somebody reading the RFC who wasn't familiar with this issue	12:44
		PM
miketheman	I'll also state like current debugging attribute level overrides is complex, this should have very clear debugging/troubleshooting to help the operator figure out what exactly they are getting.	12:45
		PM
someara	attributes in general are probably the biggest source of confusion and pain in chef	12:45
		PM
	we should be very thoughtful about changes to them	12:45
		PM
holoway	we have a spearate RFC about that	12:45
		PM
	(attributes)	12:45
		PM
thom	i think also we're going to need to do some work on ACL management, because there really isn't a good story there right now	12:45
		PM
holoway	this isn't that	12:45
		PM
	this is simply saying setting desired state is separate from reporting current state	12:45
		PM
nathenharv...	15 minute warning	12:46
		PM
holoway	so - lamont_oc, please copy-pasta this conversation in to the RFC comments	12:46
		PM

jonlives	this wins the "why wasn't it already this way" award for 2015, IMO.	12:46
		PM
holoway	and make with the typing	12:46
		PM
lamont_oc	cool	12:46
		PM
holoway	jonlives: it wins it from 2008	12:46
		PM
jonlives	heh	12:46
		PM
kallistec	For everyone interested in attributes: <a href="https://github.com/chef/chef-rfc/pull/77">https://github.com/chef/chef-rfc/pull/77</a>	12:46
		PM
thom	+1	12:46
		PM
holoway	<a href="https://github.com/chef/chef-rfc/pull/101">https://github.com/chef/chef-rfc/pull/101</a>	12:46
		PM
someara	*bites nails*	12:46
		PM
jtimberman	yay!	12:47
		PM
holoway	with the faster release cadence, I'm +1000 to adding more resources into core chef	12:47
		PM
jtimberman	i presume we'd have a separate rfc for apt?	12:47
		PM
jonlives	+1 on 101	12:47
		PM
ssd7	I think calling this thing desired and current but that being somehow different from the attribute of "observed" vs "desired" is something we need to think more about	12:47
		PM
thom	i am +1 on this	12:47
		PM
ssd7	but I'll take that to the PRs	12:47
		PM

<b>miketheman</b>	one concern I have about 101	12:47
		PM
<b>holoway</b>	My question is, do we even need an RFC	12:47
		PM
<b>jonlives</b>	in fact as LT for RHEL, I hereby appoint this on my roadmap :p	12:47
		PM
<b>thom</b>	jonlives: :D	12:47
		PM
<b>holoway</b>	I would argue jonlives can just *do* this	12:48
		PM
<b>miketheman</b>	if bugs are found, the release cycle of chef-client is slow compared to a cookbook	12:48
		PM
<b>holoway</b>	miketheman: agree - see also ^ faster release cadence	12:48
		PM
<b>btm</b>	we definitely need to work on a patch to rfc000 regarding when we need RFCs.	12:48
		PM
<b>someara</b>	my only concern his how testing is affected	12:48
		PM
<b>miketheman</b>	holoway: I'm more likely to update a cookbook than chef-client	12:48
		PM
<b>jonlives</b>	if we're generally in favor of adding more resources to core, I agree this can be delegated to LTs and maintainers.	12:48
		PM
<b>holoway</b>	someara: we can fix that by having multiple test targets so we can parallelize	12:48
		PM
<b>chip-</b>	I'm +1, especially if this eventually becomes a 'repository' resource with apt, yum, homebrew, etc.	12:48
		PM
<b>adamedx</b>	chef client release cadence will be speeding up though	12:48
		PM
<b>holoway</b>	miketheman: that's only because the cadence and test visiblity is poor	12:48
		PM
<b>chip-</b>	also, can the 'provides' functionality in 12.1 fix the cookbook vs. core resource issue?	12:48
		PM

thom	chip-: yeah, so #98 is on hold until we're happy with this one	12:49
		PM
someara	-9000 on #98 but we can talk about that later =)	12:49
		PM
miketheman	holoway: no, it's because upgrading things that change my infrastructure require forethought and reading a changelog that includes a bunch of commits unrelated to my problem will not inspire quick upgrades	12:49
		PM
thom	someara: we can figure that out :p	12:50
		PM
miketheman	(we're still on 11.16.4 since we don't have the bandwidth to "prove" chef 12)	12:50
		PM
holoway	miketheman: this is my point - that's because 12 isn't safe	12:50
		PM
	you don't trust the release process, and you know it will break you	12:50
		PM
	so you stay old	12:50
		PM
miketheman	I think it's also unrealistic to expect "just grab latest from nightlies, and you'll be fine"	12:50
		PM
holoway	it's a rational choice	12:50
		PM
miketheman	:)	12:50
		PM
holoway	how often do you upgrade chrome?	12:51
		PM
jonlives	miketheman: I think that's more "grab the latest if you really really need this bugfix"	12:51
		PM
nathenharv...	10 minute warning	12:51
		PM
jtimberman	jonlives: precisely	12:51
		PM
holoway	answer: all the time, because it's super safe	12:51
		PM

adamedx	yeah	12:51
		PM
jonlives	fwiw, we're still on 11.16 too, working on 12 but not there yet.	12:51
		PM
miketheman	so moving resources into core chef need to be battle-hardend and fully amazing	12:51
		PM
adamedx	and you can canary the nightly	12:51
		PM
cheeseplus	holy scrollbar	12:51
		PM
adamedx	to see if you want to deploy it more widely	12:51
		PM
martinisoft	We're still on 11.16.4 for similar reasons, though we need to do more Chef 12 compat testing before moving up.	12:51
		PM
holoway	but right now, you can't do that	12:51
		PM
jonlives	miketheman: I'd argue that also kinda makes changing existing resources hard, ie the multi-package stuff that went into the package provider.	12:51
		PM
adamedx	taking note of your 11.* versions miketheman and martinisoft	12:52
		PM
holoway	so I'm cool with saying moving more resources into core is dependent on release cycle	12:52
		PM
someara	maybe we should adopt a DSC style semantic naming thing... xYum_repository or something	12:52
		PM
jtimberman	this is why the version specification in kitchen.yml is super good.	12:52
		PM
someara	just thinking out loud	12:52
		PM
adamedx	not unreasonable	12:52
		PM
martinisoft	I think RFC 101 is on the right track to pull more common cookbooks into core, +1 from me	12:52
		PM
adamedx	kind of aligned with feature flags	12:52



		PM
<b>miketheman</b>	jonlives: yes, which is a "scary" change	12:52
		PM
<b>holoway</b>	but the first use experience of digging about for the right cookbook is quite shitty	12:52
		PM
<b>adamedx</b>	someara	12:52
		PM
<b>miketheman</b>	adamedx: sure thing. :)	12:52
		PM
<b>martinisoft</b>	Being a primarily (right now) Ubuntu stack right now, it has become muscle memory to include the 'apt' cookbook.	12:52
		PM
<b>holoway</b>	tl;dr - I believe you have stockholm syndrome	12:53
		PM
<b>adamedx</b>	always include apt	12:53
		PM
<b>holoway</b>	right	12:53
		PM
<b>miketheman</b>	holoway: I'm not against moving resources into core chef - "batteries included" - but we have to balance the amount of change to these resources vs the release cycles	12:53
		PM
	<i>someara hatches evil plan to move All The Cookbooks into core</i>	12:53
		PM
<b>jtimberman</b>	adamedx: and the "windows" cookbook :l	12:53
		PM
<b>martinisoft</b>	Yeah, but why can't that just... like be there for a platform? :)	12:53
		PM
<b>holoway</b>	if they move into core, what they inherit is the stability constraints of core	12:53
		PM
<b>adamedx</b>	we've already moved half of it to core jtimberman :)	12:53
		PM
<b>jonlives</b>	holoway: it's sort of stockholm syndrome where the kidnappers will fix your issues quicker than the hostage negotiators ;)	12:53
		PM
<b>jtimberman</b>	adamedx: ha! lets get the other half in	12:53

		PM
<b>adamedx</b>	yes	12:53
		PM
<b>holoway</b>	that's the thing	12:53
		PM
<b>martinisoft</b>	See also: All the other Ubuntu platform supported Chef (formerly Opscode-Cookbooks) cookbooks	12:53
		PM
<b>holoway</b>	you can't be a cowbody if its in core	12:53
		PM
<b>miketheman</b>	holoway: I hope someara knows he can't be a cowboy today already	12:54
		PM
<b>holoway</b>	:)	12:54
		PM
<b>someara</b>	giddyup	12:54
		PM
<b>holoway</b>	okay - I think this doesn't need an RFC	12:54
		PM
	we should do it, but I'm moving this to the LTs	12:54
		PM
	if you run ubuntu, rhel, freebsd	12:54
		PM
	you have my fiat blessing to move resources into core	12:54
		PM
<b>jonlives</b>	I think it makes sense for component LTs to be able to deciderate adding resources to core, but they don't ger merged without good test coverage, code review etc etc	12:54
		PM
<b>miketheman</b>	it needs to be trumpeted from the rooftops, as well as "how does chef-client handle a case where a cookbook has the same resource"	12:55
		PM
<b>holoway</b>	jonlives: sure - it has to be good code, which is also the LTs and maintainers responsibilty	12:55
		PM
<b>ssd7</b>	+1 there	12:55
		PM
<b>jonlives</b>	so again, what holoway said while I was typing. I need to speed up. :p	12:55

		PM
<b>martinisoft</b>	jonlives: Agreed there, this is why we have the core LT's yes?	12:55
		PM
<b>ssd7</b>	I really don't want on RFC for every minor feature or resource	12:55
		PM
<b>nathenharv...</b>	5 minute warning. I move that we push all remaining PRs to next week's meeting	12:55
		PM
<b>holoway</b>	the core doesn't need to be involved	12:55
		PM
<b>martinisoft</b>	Can be proposed for merge to core with the LT for core having sign off	12:55
		PM
<b>thom</b>	this sounds like goodness to me	12:55
		PM
<b>holoway</b>	you - martinisoft - can just merge anything you think is right for freebsd users	12:56
		PM
	assuming your maintainers agree	12:56
		PM
<b>jonlives</b>	I have deciderated my first RFC, hurray :p	12:56
		PM
<b>holoway</b>	:)	12:56
		PM
<b>btm</b>	ssd7: I've got a personal card to propose changes to RFC000 regarding what needs an RFC.	12:56
		PM
<b>someara</b>	miketheman the cookbook should "win" I think... monkey patching chef classes in a cookbook is a decent way to to development	12:56
		PM
	<i>martinisoft crafts a big red merge button for his desk</i>	12:56
		PM
<b>miketheman</b>	someara: let's validate that and ensure it works forever	12:56
		PM
<b>holoway</b>	you don't need permission from Thom (the core LT) or me	12:56
		PM
<b>miketheman</b>	(or until chef 13)	12:56
		PM

<b>holoway</b>	right	12:57
		PM
	back-compat here is a thing	12:57
		PM
	so if we want it in 12	12:57
		PM
	we have to figure out how ot not break the world	12:57
		PM
<b>someara</b>	miketheman can you annotate that in the PR?	12:57
		PM
<b>holoway</b>	and simply saying "upgrade your cookbooks" is not enough	12:57
		PM
	:)	12:57
		PM
<b>btm</b>	someara/miketheman: that changed with Chef 12 (see windows_package LWRP having to become a library).	12:57
		PM
<b>miketheman</b>	holoway: you win a teddy bear!	12:57
		PM
<b>jonlives</b>	so do we want to merge 101 as an RFC, or just wait for a PR to chef from someara	12:57
		PM
<b>holoway</b>	there is no RFC	12:57
		PM
<b>jonlives</b>	cool, will close that off then.	12:57
		PM
<b>holoway</b>	jonlives can simply decide	12:57
		PM
<b>martinisoft</b>	there is no spoon, either. What is left in the agenda?	12:58
		PM
<b>adamedx</b>	chef update	12:58
		PM
<b>holoway</b>	but we're all being adults about back-compat - so you're a bad LT if you break all the RHEL ecosystem in your zeal	12:58
		PM

	:)	12:58
		PM
<b>miketheman</b>	is there a punishment for that?	12:58
		PM
<b>holoway</b>	nope!	12:58
		PM
<b>chip-</b>	miketheman: I'm pretty sure you can use the dynamic provider to have the cookbook win in case of resource name collision: <a href="https://www.chef.io/blog/2015/02/10/chef-12-pro...">https://www.chef.io/blog/2015/02/10/chef-12-pro...</a>	12:58
		PM
<b>holoway</b>	hugs and a blameless post-mortem	12:58
		PM
<b>jonlives</b>	miketheman: the punishment will be that I use CentOS exclusively, so I'd be causing myself great pain	12:58
		PM
<b>holoway</b>	exactly	12:58
		PM
	<a href="https://github.com/chef/chef-rfc/pull/103">https://github.com/chef/chef-rfc/pull/103</a>	12:59
		PM
<b>miketheman</b>	awesome	12:59
		PM
<b>github-bawt</b>	[chef-rfc] jonlives closed pull request #101: adding add-yum_repository-to-core.md (master...add-yum_repository-to-core) <a href="http://git.io/pvRb">http://git.io/pvRb</a>	12:59
		PM
<b>someara</b>	btm can you throw some details/notes about that in the 101 PR?	12:59
		PM
<b>holoway</b>	I really want to talk about this with out remaining minutes	12:59
		PM
	mailing list migration failed	12:59
		PM
<b>btm</b>	someara: yes	12:59
		PM
<b>someara</b>	thanks	12:59
		PM
<b>jtimberman</b>	regarding cookbook resources vs chef core resources, homebrew is an example of this.	12:59
		PM

holoway	I think saying we move to discourse and stackoverflow is actually healthy for us	12:59 PM
miketheman	+1 on discourse	1:00 PM
martinisoft	I like discourse	1:00 PM
jonlives	there are only so many choices here, anyway	1:00 PM
holoway	we know there will be people who prefer email - and they can subscribe via discourse	1:00 PM
	<i>kallistec waves</i>	1:00 PM
someara	I have no opinion on the matter so long as it's still a "mailing list" with an smtp interface	1:01 PM
thom	this is one of those that i have no strong opinion on besides "email turns up in a timely manner"	1:01 PM
	someara: ^5	1:01 PM
miketheman	thom: such demeands. ;)	1:01 PM
holoway	sure	1:01 PM
cheeseplus	as anecdotal date from someone manning the IRC helm often: I've seen so many people asking questions in IRC but then posting dupe stackoverflow issues but afraid of mailing lists	1:01 PM
holoway	yep	1:01 PM
	I think we need to bless stackoverflow too	1:01 PM
miketheman	it certainly doesn't help when mailing lists get mean	1:02 PM
holoway	ok - I have another meeting	1:02 PM

<b>someara</b>	I should mention that I registered <a href="http://chefoverflow.com">chefoverflow.com</a> a few years ago	1:02
		PM
<b>miketheman</b>	there's also no "reputation" to be gained on mailing lists vs SO/SF	1:02
		PM
<b>chip-</b>	Did we get enough high-ranked questions answered for our own stackoverflow?	1:02
		PM
<b>holoway</b>	I don't think so	1:02
		PM
	but I don't think it matters	1:02
		PM
<b>thom</b>	but we see the other side of that, which is people posting on SO and then on the ML when they don't get a response. So maybe we ought to say "SO is the go to point for questions" and then have a single mailing list for devel-y stuff	1:02
		PM
<b>holoway</b>	we can bless the tag	1:02
		PM
<b>nathenharv...</b>	chip-: <a href="http://area51.stackexchange.com/proposals/77609">http://area51.stackexchange.com/proposals/77609</a> ...not there yet	1:03
		PM
<b>miketheman</b>	we're overtime. next week?	1:03
		PM
<b>holoway</b>	yes	1:03
		PM
	next week	1:03
		PM
<b>adamedx</b>	Chef Update: we released Chef Client 12.1.1 with fixes for some 12.1.0 regressions (including broken patchkage providers) on Monday	1:03
		PM
	focusing on increasing our release frequency for Chef Client to be more like sdelano and legion of builds	1:03
		PM
	that's it	1:03
		PM
	great community fixes in that release as well — thank you	1:04
		PM
<b>btm</b>	nathenharvey: having a 20 minute "roundtable" on the agenda's seems super wishful. Can we "allocate" more time for PRs and not "allocate" time for the	1:04
		PM

roundtable on future agendas?

**nathenharv...**

\*\*\* MEETING ENDS \*\*\*

1:04

PM