1. Introduction
	1. It’s Saturday afternoon after a long week, so my presentation is going to be light on the text, and hopefully heavy on entertainment.
	2. Rockefeller Archive Center is an independent operating foundation that preserves and makes available for research the archival collections of members of the Rockefeller family, institutions and organizations founded by Rockefeller family members (including the Rockefeller Foundation, Rockefeller Brothers Fund, Winthrop Rockefeller Foundation, General Education Board, Rockefeller University, Population Council, Asia Society, and many other organizations).
	3. About twenty FTE archivists working with the system.
	4. ArchivesSpace implementation team was essentially three people: myself, my supervisor (Sibyl Schaefer), and one of our sys admins (Jose Morillo).
	5. We had 13,944 name records, 8,246 subject records, 3,264 accession records, 11,304 resource records, and 2,401 digital object records.
	6. The entire process took about 500 hours of my time. I had some experience with MySQL, python, and linux beforehand, but I did have to learn a lot as I went along. Others could probably finish faster.
	7. We were migrating from AT to AS, and we had all of our finding aids already into the system.
	8. We have been live with ArchivesSpace since December 2014, but we do not use the public display portion of the site.
	9. We are hosting our own system on an Apache Virtual Machine, and we have two servers up and running (one for production and one for development)
2. Challenges
	1. I’ll now jump into the major challenges we ran into while preparing for migration. All in all, they weren’t so bad, but they do take time.
		1. Number one issue that I spent the most time on: data cleanup before the migration.
		2. We had, and in some ways still have, very messy data.
		3. Probably the easiest issues to fix were the controlled value lists (extents, containers, instances, etc.). We had so many duplicate entries that I think at one time we had about 3 different entries all for describing a box (Box, Boxes, Box(es)), and this was not an isolated instance. Here is a list of our extent types:
			1. bound volume
			2. Box
			3. box(es)
			4. Cubic feet
			5. cubic ft.
			6. document box
			7. document boxes
			8. document cases
			9. feet
			10. film reel
			11. Folder
			12. Folders
			13. Hanging folders
			14. item(s)
			15. Linear feet
			16. microfilm reels
			17. pages
			18. record cartons
			19. reels
			20. tapes
			21. Volume
			22. volumes
		4. These duplications ran throughout our agents and subjects as well, we would sometimes have the same subject in three times, once as a LCSH topic, once as an LCSH geographic term, and once as a local term, with no variation.
		5. We knew that we wanted to clean as many of these up as possible, so I undertook a large effort to identify duplication in subjects and agents in the backend AT database and then merged them in AT prior to migration.
			1. You can do this with some fancy SQL commands
		6. Adding a note here at the end: You can avoid most of these by having well-defined documentation and description policies. One thing that came out of this work was the impetus to revise our current description manual to be more explicit about terminology.
		7. Should have taken a closer look at locations (we are dealing with issues right now)
	2. Migration testing
		1. Leave yourself plenty of time to do test migrations.
		2. Our first migration test threw back 130 errors. Many because of duplicate terms I had missed somewhere, but some for various other data reasons.
		3. Doing a migration test can be a helpful way of figuring out where you might have data issues before you start cleanup.
		4. To put it bluntly, when we first were testing the migrations, the migration tool did not have the most helpful error reporting.
			1. “Status code: 400. Status text: Bad Request. {“error”:null}. Anyone know what that one means? I still don’t, but with the help of the ArchivesSpace user group, we tracked it down to actually being a migration tool issue, and not a data issue.
		5. Ultimately, iterative testing is absolutely necessary.
	3. Troubleshooting Errors
		1. AS is still fairly new and does have some bugs.
		2. One of the greatest challenges was using the system before our archivists and understanding where issues arose and why.
		3. It’s important to have more than one person working with the system and testing it, because different people work with it in different way. We had a very small closed testing period in order to identify any issues with the system.
		4. Unfortunately, sometimes these issues will persist, and you will have to either find a way to deal with them, or wait for the next release or patch until they get fixed.
	4. Training
		1. Training is one of the biggest areas of any new system, and it can vastly impact user adoption.
		2. I spent 20 hours meeting with every single person that would be using the system and tailored training for specific groups (reference, description, accessioning).
3. Successes
	1. Full data transfer…almost. We had three VERY large test resources in AT that were so large they didn’t migrate because the tool ran out of java heap space no matter how much memory we gave java. They were about 10 megabyte EAD files each, maybe larger.
		1. Otherwise, no missing data.
	2. User adoption has been great
		1. Quick transition from AT to AS, and I don’t think archivists would complain, except for a few bugs (component reordering).
	3. New opportunities through API
		1. Much more power and control over what you want to get out of the system
4. Looking forward
	1. Automatic export of EAD and PDFs whenever a user makes a change to a resource (only exports published data
		1. Integrating AS with Github EAD repository
	2. Integration with Archivematica to handle automatic creation and pairing of Digital Objects to archival components
	3. Automatic rights management control handled through PREMIS in AS