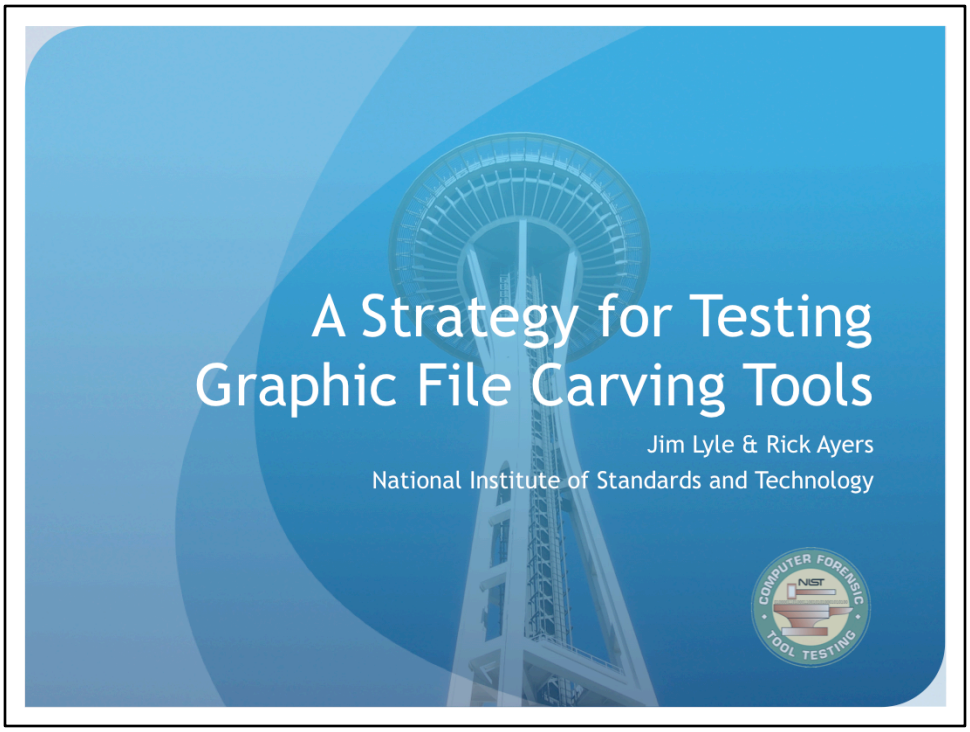
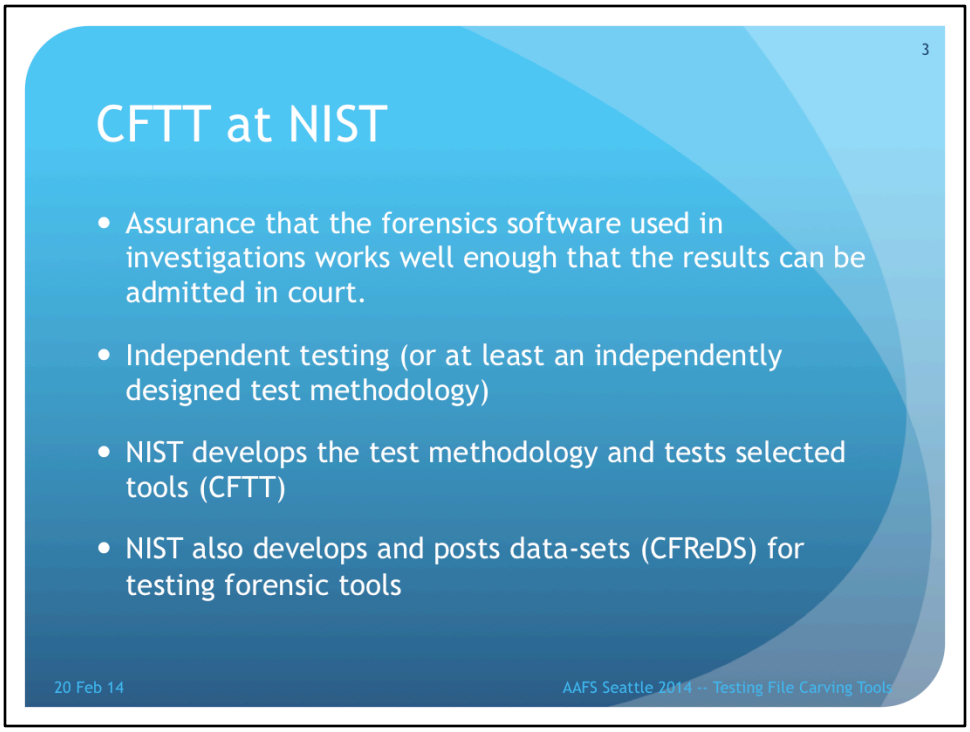
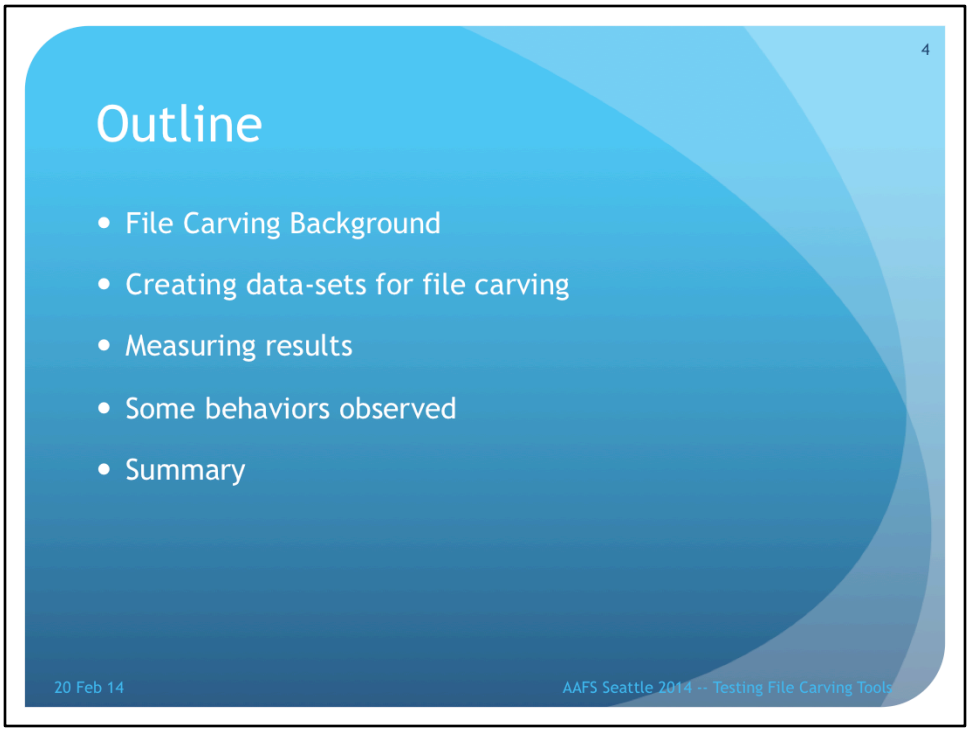



Thank you for coming to my talk today. I'll try to be informative about testing file-carving tools at NIST.

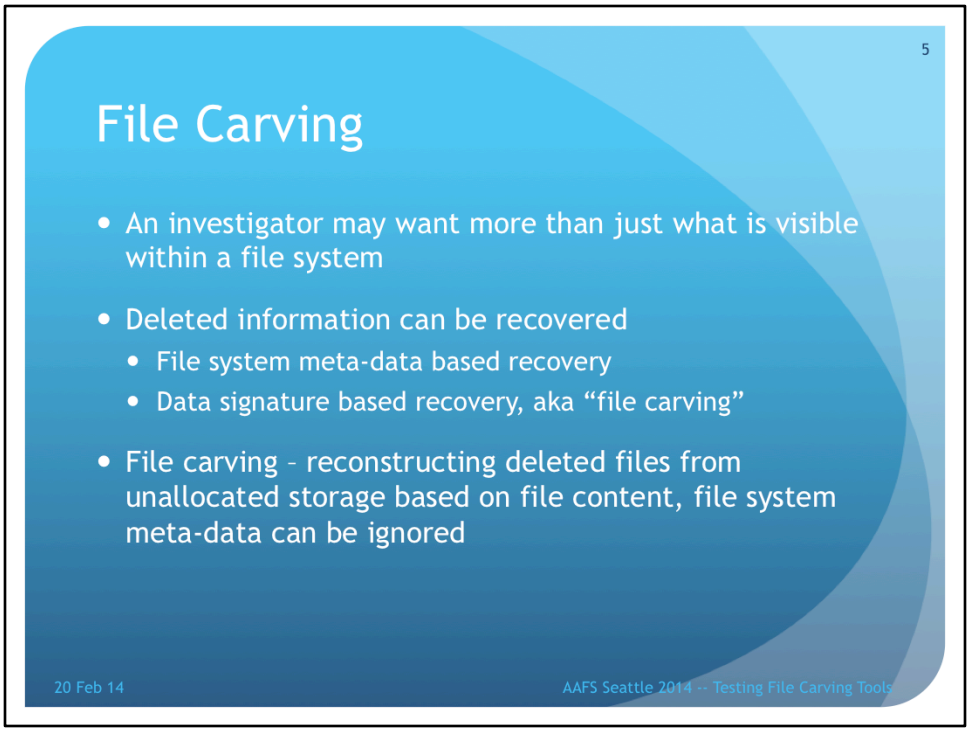

This is the standard NIST disclaimer. I may mention any available file-carving tool, either commercial or freely available. This includes any of the major forensic tools used for digital evidence.



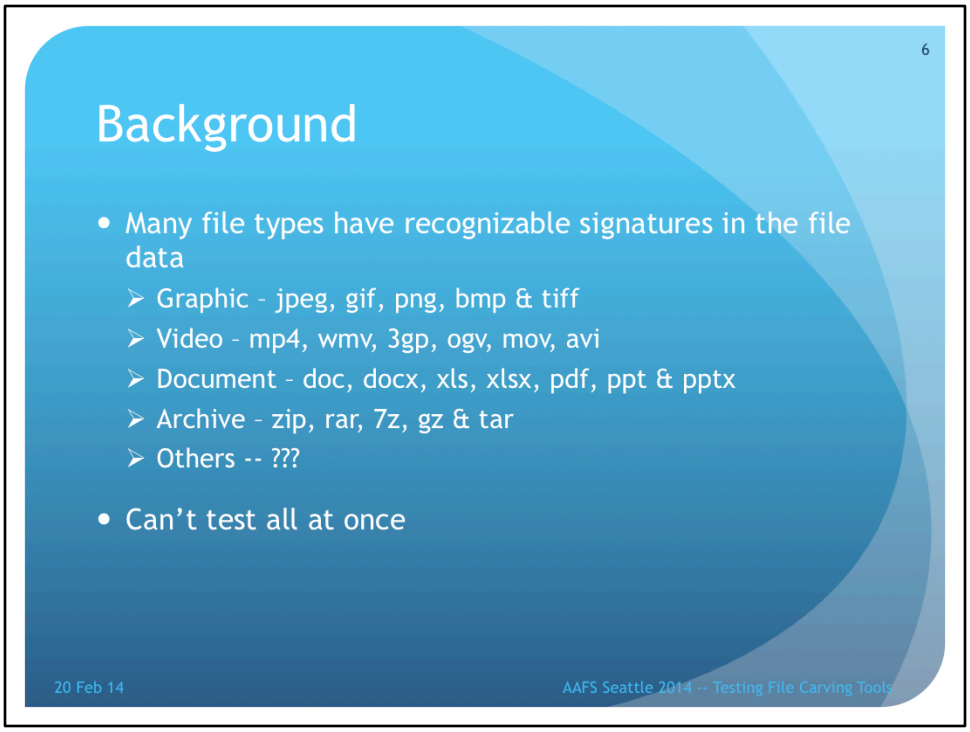

The aim of the CFTT project at NIST is to provide assistance to the digital forensics community by developing methods for testing forensic tools, by testing forensic tools, and by making test data sets available to the general community.



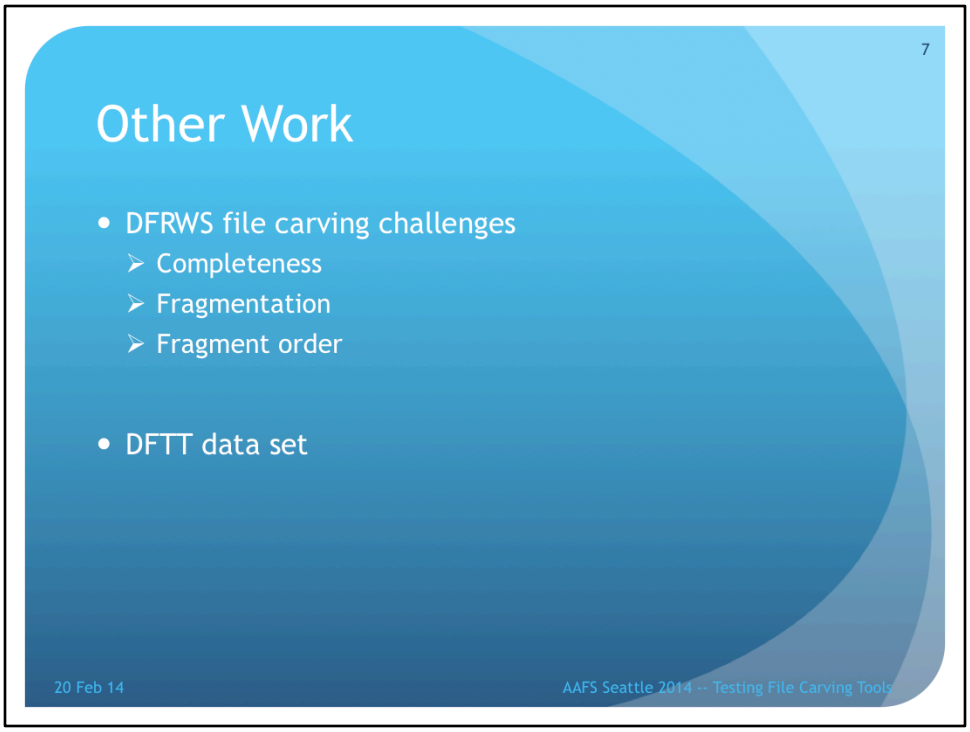

Today I'm going to talk a little bit about testing file-carving tools NIST.
First I'm going to talk about how we created data sets second I'll talk about approaches to measuring the results obtained, and finally I'll discuss some of the behaviors we've observed.



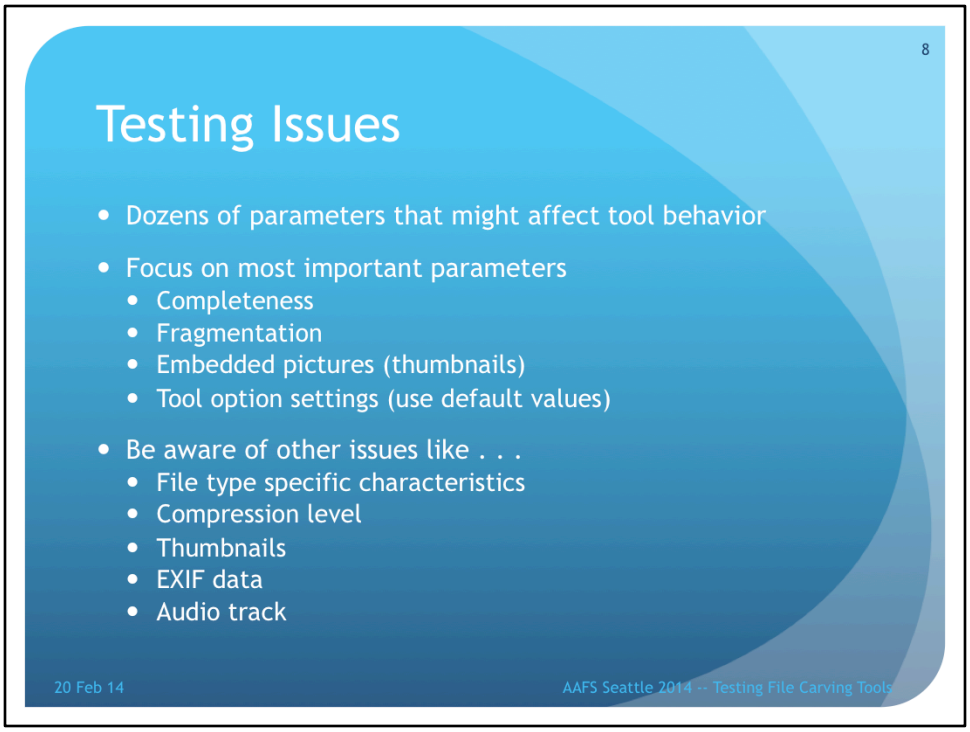

There's more useful information to be found than just what's visible walking the file system. Deleted files may be recoverable either from metadata or from an examination of the actual data and looking for file signatures. What I'm talking about today is file carving that is reconstructing deleted files from unallocated storage based on the file content.




Not all file types have recognizable signatures for example a plaintext file doesn't have a signature, but might be identified by analysis of byte frequencies. However other files specifically pictures, videos and MS Office documents have a very definite structure that's easy to recognize. It would be rather daunting to try to test in one test report all these different types of files being carved so rather than try to test all of them at once we decided to break your testing up into groups. These groups seem rather natural to use. For example with the graphics files we can test all of the tools the carve that kind of file once and not worry about other types of objects.

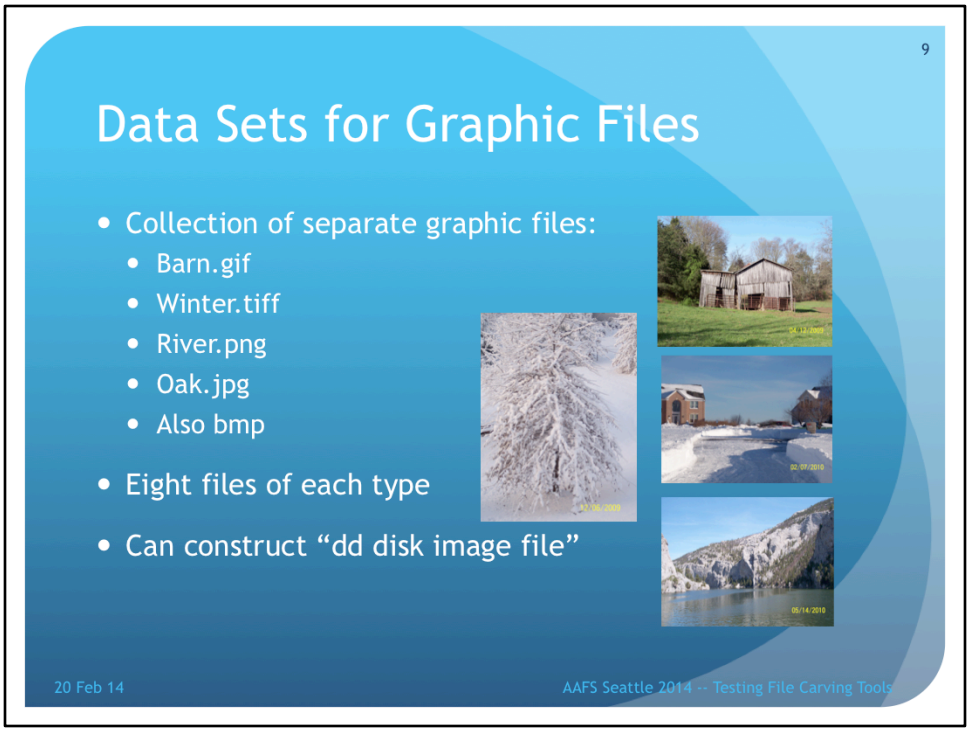



We didn't want to start from scratch. We looked at other available file carving test data. The main features used in the DFRWS file carving challenges were having complete images, having fragmentation and having fragmentation out of order. This seemed like a good place to start.

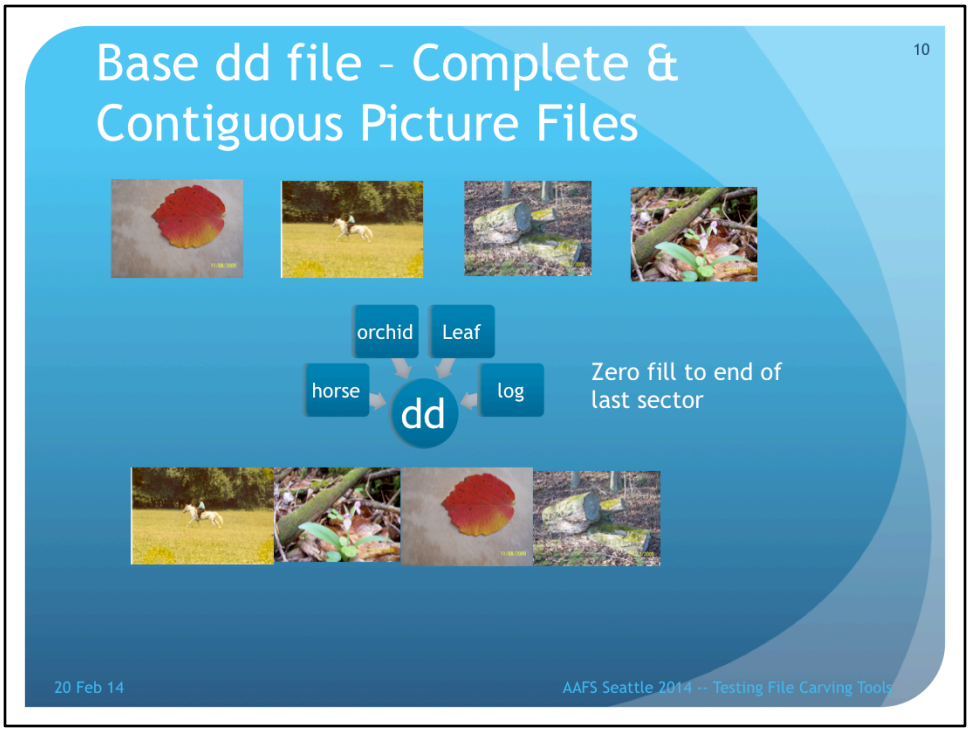



Constructing test data for software is always a challenge. You have to put in enough features to reveal any interesting tool behaviors but you have to balance that with not trying to do too much at once. You want to have some easy things, like complete contiguous files, for the tool to do. This calibrates that you got the tool working and it's doing some basic recovery operations. Then you can add on some more interesting features to the data so that you can see how good the tool is with more challenging data sets like fragmented files or incomplete files.

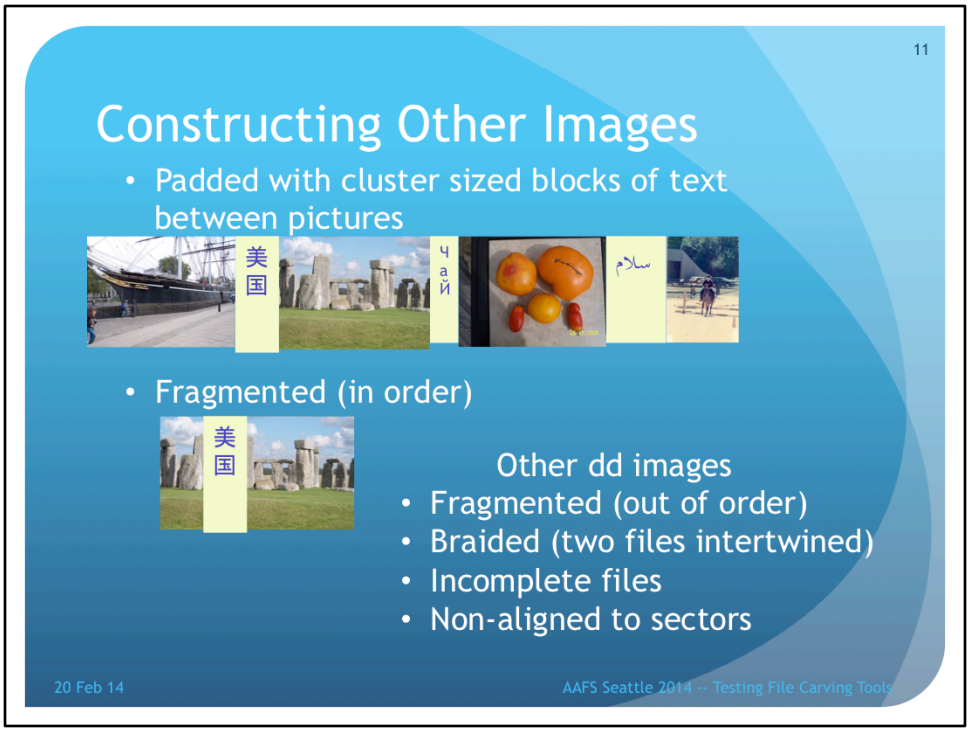
Tool options are another challenge, trying every possible option and combination of options is not feasible. We could wind up with a large number of test runs. So we just hope that the vendor puts his best set of options as the default and we then use the default. Of course this isn't always going to be true but it gives us a uniform way to treat all the tools.
There are other issues that we are aware of but we didn't include in this set of test data. We may come back in a later version of the test data and add some of these things in. These can be important, for example for one tool, if the first data in the dd file is exif data, like in a JPEG file, the tool doesn't carve any thing.



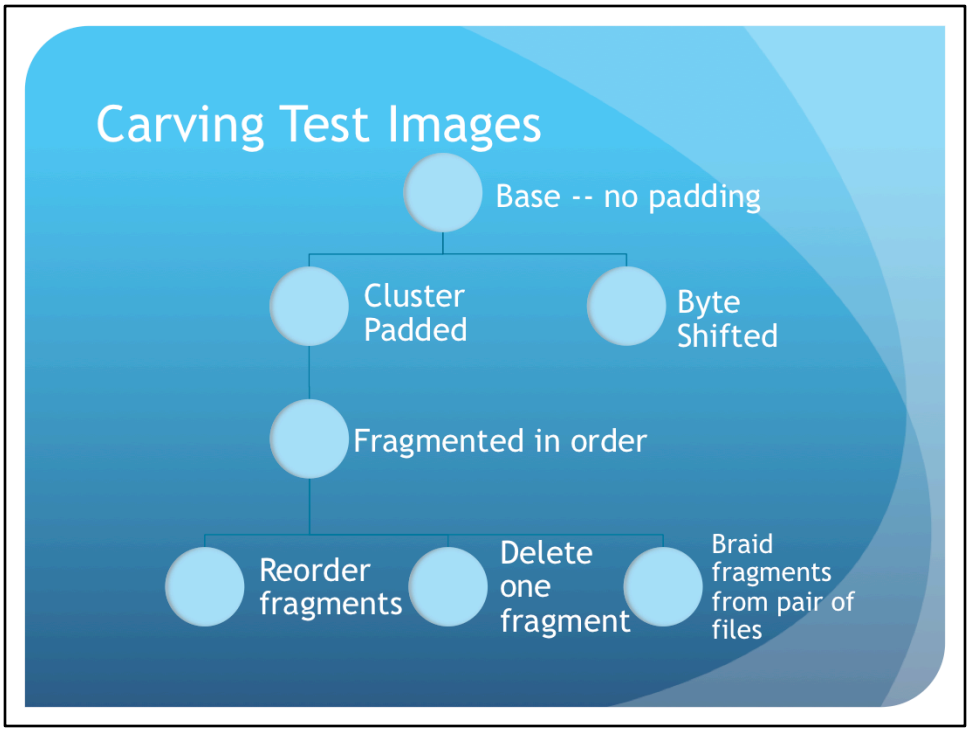

The test data we are using is a set of dd disk image files that might be created by capturing the unallocated space from the hard drive. The source data for the DD files is a set of 40 graphics files, five file types, eight files of each type. To avoid confusion I will try to refer to disk image files that you might get by acquiring a hard drive as dd files and picture image files as graphics files or by file type, like jpeg.



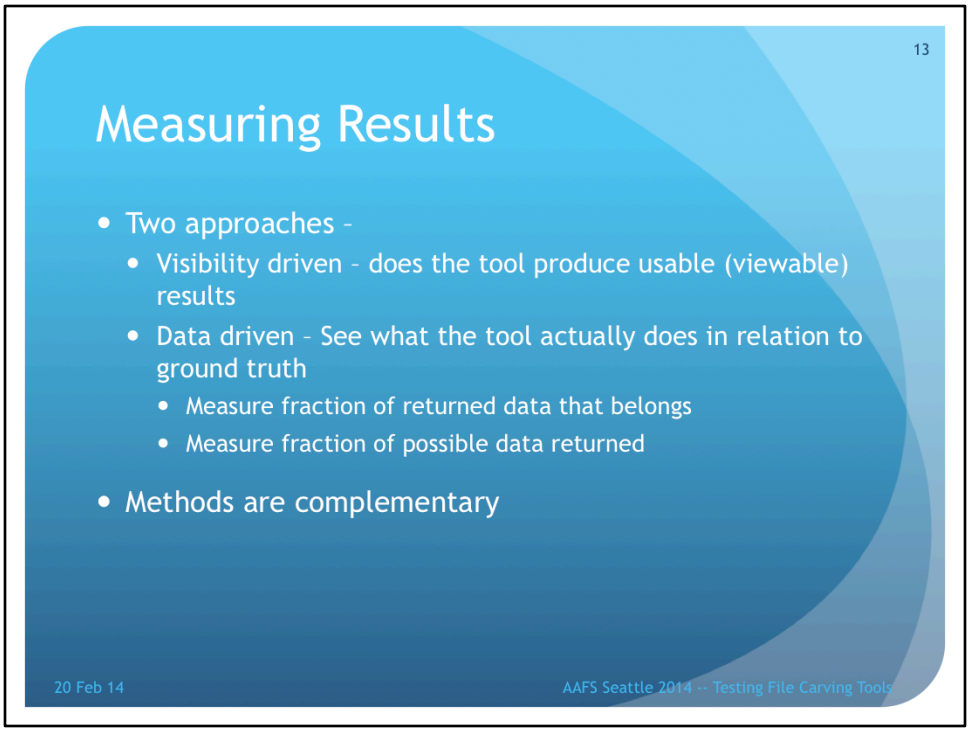

We don't actually image unallocated space. What we really do is concatenate all 40 files together into one file and that is  our first DD file. We also want to ensure that all files begin on a sector boundary. If the last sector doesn't take up 512 bytes we add zero value bytes to make the file not have any file slack. This gives us a base image with complete contiguous files starting on sector boundaries that should be the easiest task for a file carver to work on.



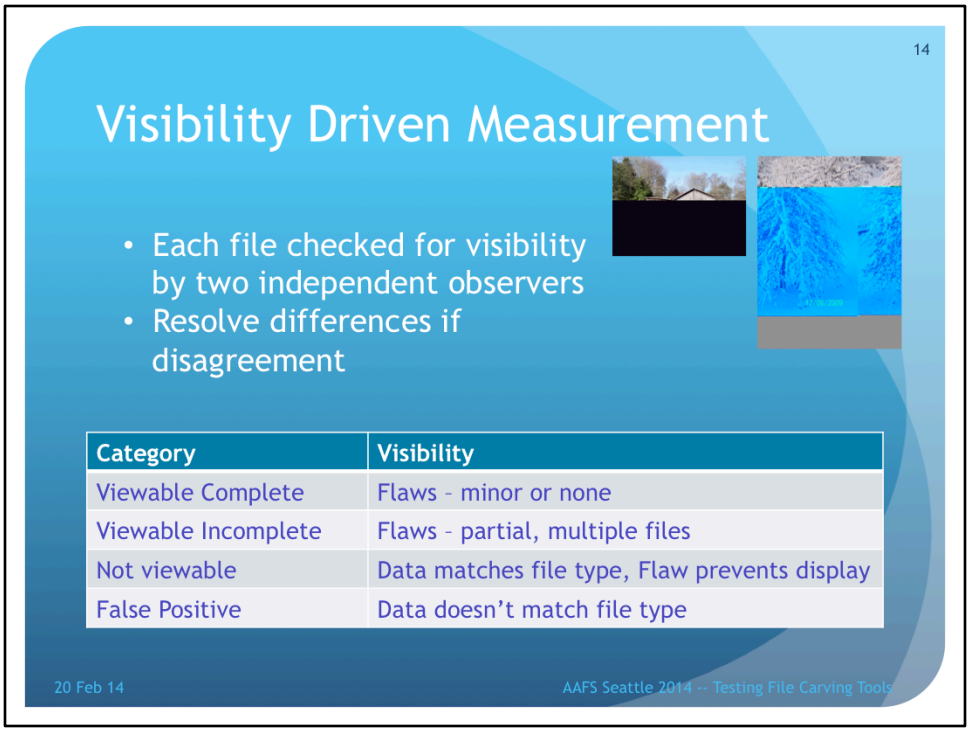

For other DD files we do the following: complete contiguous files with cluster size padding between each file. The content of the padding varies. Some padding is just zero bytes, other padding is random bytes, we also use text. Some of the text is English ASCII other text is foreign-language text and Unicode UTF-8 Unicode 16 ASCII upper bit standards for languages like Arabic and Russian and also text encoded in base 64 and uuencode.



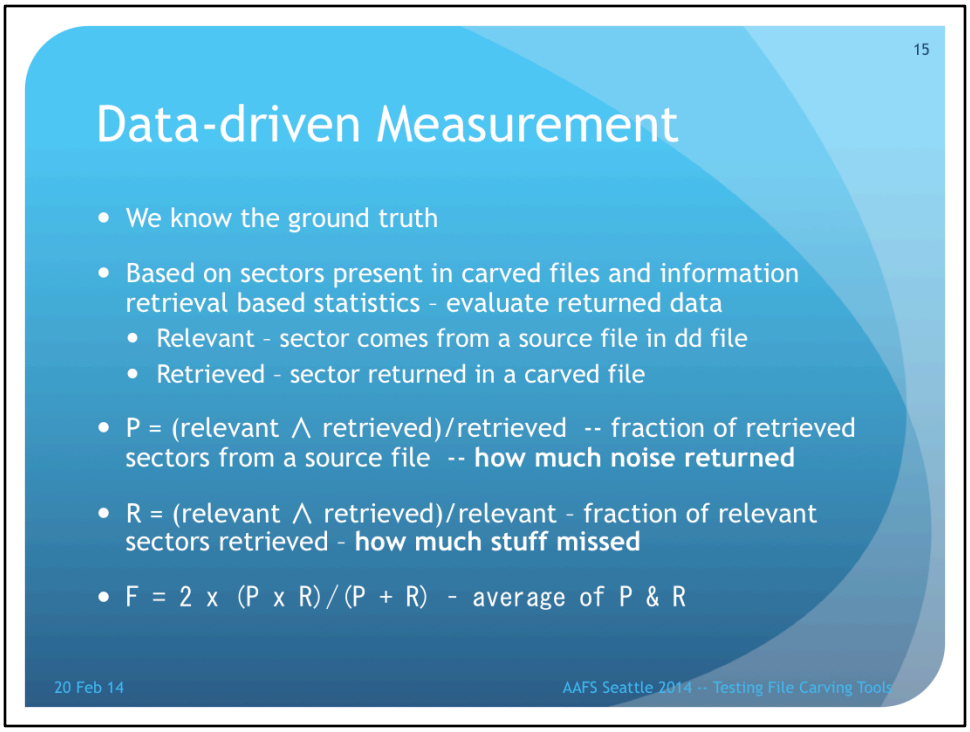

Here are our current image files. Starting with the base file of just images concatenated together. Then we have one image where the graphic files are separated as if there's other text files between. We separate the graphic files with cluster sized chunks. The second image has padding of just a few bytes so that the graphic file signatures are not aligned on sector boundaries. The other four images are variations on fragmentation and completeness.




Now that we have our DD files we are ready to run some tests. We run the tool we want to test. We attach the DD test file we want to use. Then we see what the tool does. After the tool finishes carving we export all the files that are carved. So how do we measure what the tool has done? We know all the ground truth of what could be recovered and we have what was recovered so it is easy to look through the data and see if we miss anything or if we get something extra. But the practitioner really cares the most about if the tool produces visible graphic files that can be used as evidence. We use two measures of tool behavior. First we look at the visibility of the return files and second we look at the actual data returned so in addition to knowing how much we can see, we know how much data is actually returned and how much is missed.



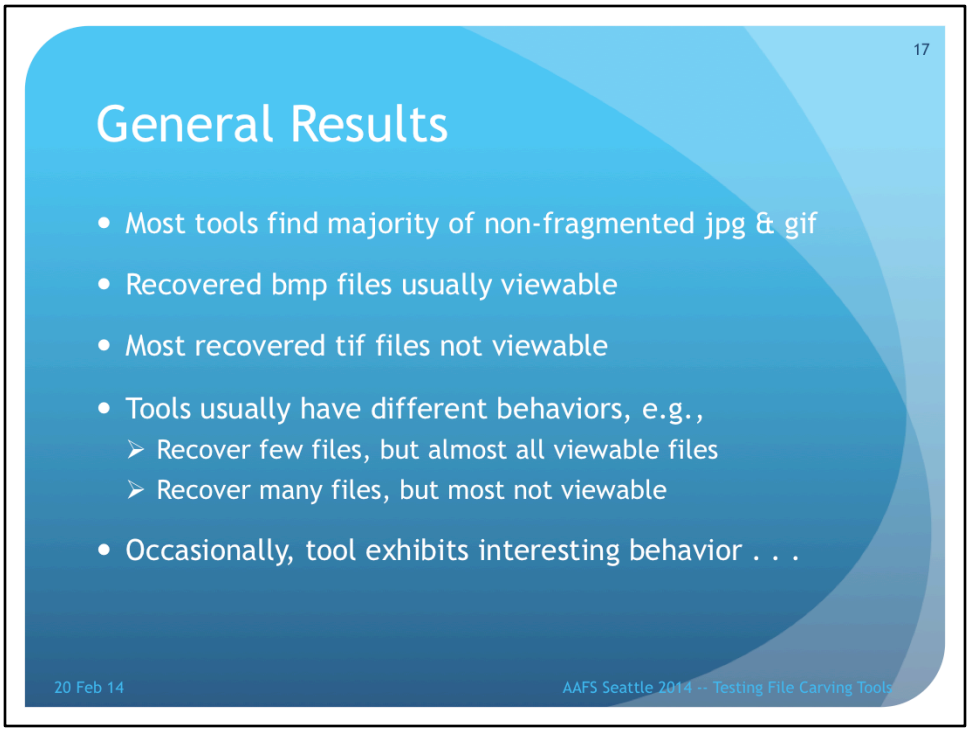

We thought of lots of ways to classify the visual results. We decided to go with a very simple classification of basically four categories. Did we get most of the file such that you can figure out what the file is. We call that visible complete. Sometimes we get a chunk of a graphic file and we can tell a little bit about the file but we don't see the whole picture. Third category is for files that we can't tell what it is or maybe it doesn't even display at all. The fourth category is for files that when we look at the file extension like a JPEG but then the content doesn't come from a JPEG file it's something else. The tool decided to declare this is a JPEG file even though it may be from a tiff file.



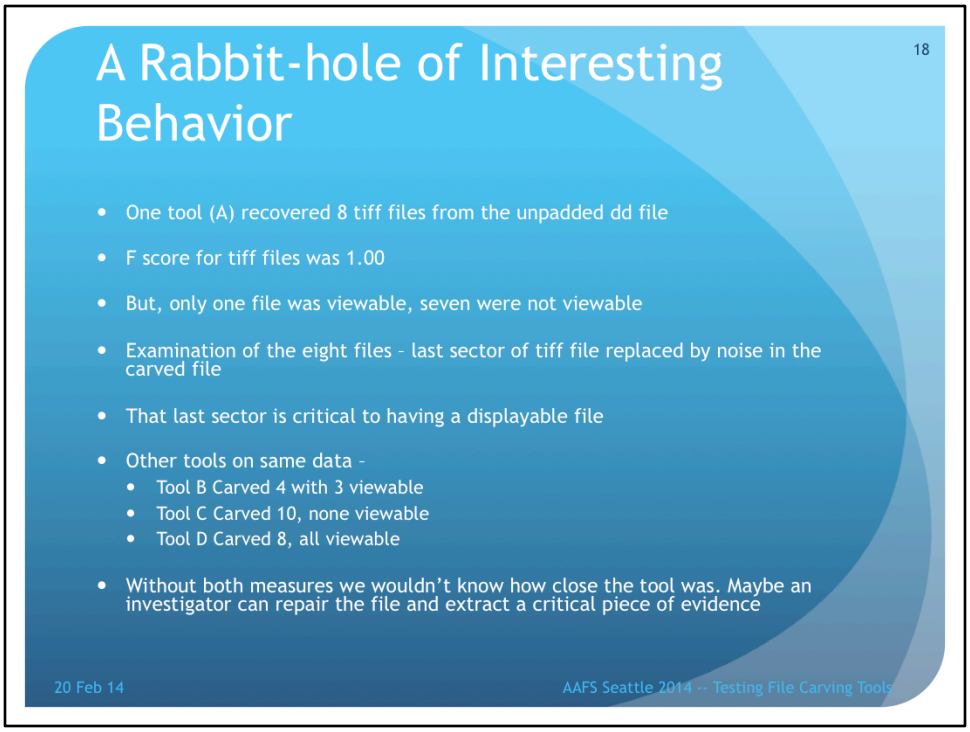

A data driven measure gives us a good idea about how much data is missed and how much stuff is actually noise coming from somewhere else that doesn't really belong in the graphic image file. I'm not going to say more on this but you can see it's based on information retrieval type metrics.


Currently here's our plan. We have run tests on about a half-dozen file carvers on graphics files the reports are in the final stages of our NIST review process and should be published very soon. We are testing both general-purpose tools and specialty tools.

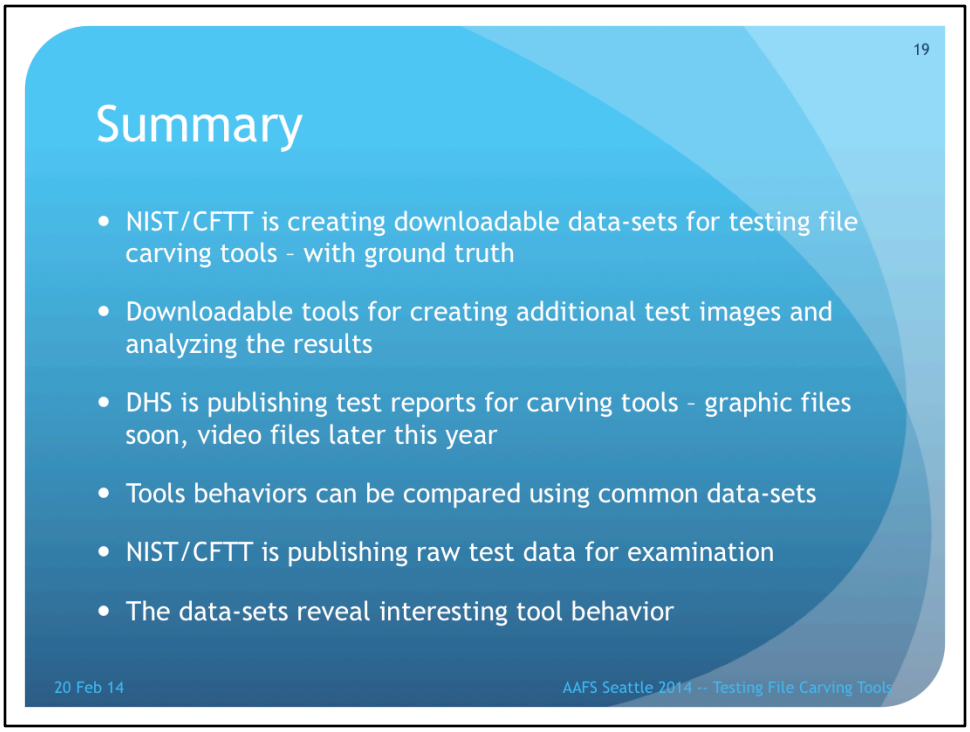
We have just finished running our test cases for video files. We are drafting the reports right now and will begin the review process soon. So where do we go next? we could look at carving for documents for archived files maybe audio will start that pretty soon.




In general the tools do pretty good for the file formats they really support. Some tools do not support all the file formats we tried and so they don't produce anything for them. Tiff files seem to be one of the more difficult formats to carve.




There are lots of rabbits that can be chased down. For the tool I'll call a we noticed that for the data-driven measurement of tool behavior we got great results. our measure was 1.00 which pretty much means perfect behavior no noise all the data is recovered. But, only one file was viewable. Something's interesting here, maybe we made a mistake. On a closer examination of what was returned it turned out that the last sector was missing from the TIF files. This is where having two complementary measures really helps. If we only looked at file visibility we would conclude of this tool doesn't give us anything. But after we looked at the data that was recovered and see the last sector is missing this gives us the potential for trying to repair that last sector or maybe go into the original file was carved from and manually pull out that last sector and attach it and then you got everything.
Three other tools gave rather mixed results I like tool D it looks like my favorite for carving TIF files.




So far this makes a nice contribution for practitioners that want to test their carving tools. We got testator that we've made avail available through our website that can be downloaded and used to test carving tools. We also plan to make available tools we used for creating the DVD files and also the analysis tools used to look at the results. The raw test result data will also be made available this can be mined for interesting additional results.


20
Sponsors
• NIST OLES
• DHS S&T
20 Feb 14
AAFS Seattle 2014 -- Testing File Carving Tools

# Contact

Jim Lyle
JLYLE@NIST.GOV

Rick Ayers
RICHARD.AYERS@NIST.GOV

Barbara Guttman
BARBARA.GUTTMAN@NIST.GOV

Susan Ballou
SUSAN.BALLOU@NIST.GOV

http://www.cfreds.nist.gov — Test Data Sets

http://www.cftt.nist.gov — Test Reports

# Thanks, Any Questions?