Data Flow
Data Simulators
Data Challenges

Stephen Bailey

Lawrence Berkeley National Lab

DESI Data Systems Lead

Argonne 2015

Why bother with integration now?

Science readiness

- Science requirements are different than operations requirements
- We need to support both

Interfaces are often harder than you think

- Actually putting the pieces together reveals unanticipated issues
- Don’t wait until 2019 to integrate

Usability

- If you don’t like how it works, let’s fix it now

Work in Progress

Mix of

- Pretty good
- Partially done
- Unimplemented ideas
- Completely missing

Don’t be shy about feedback

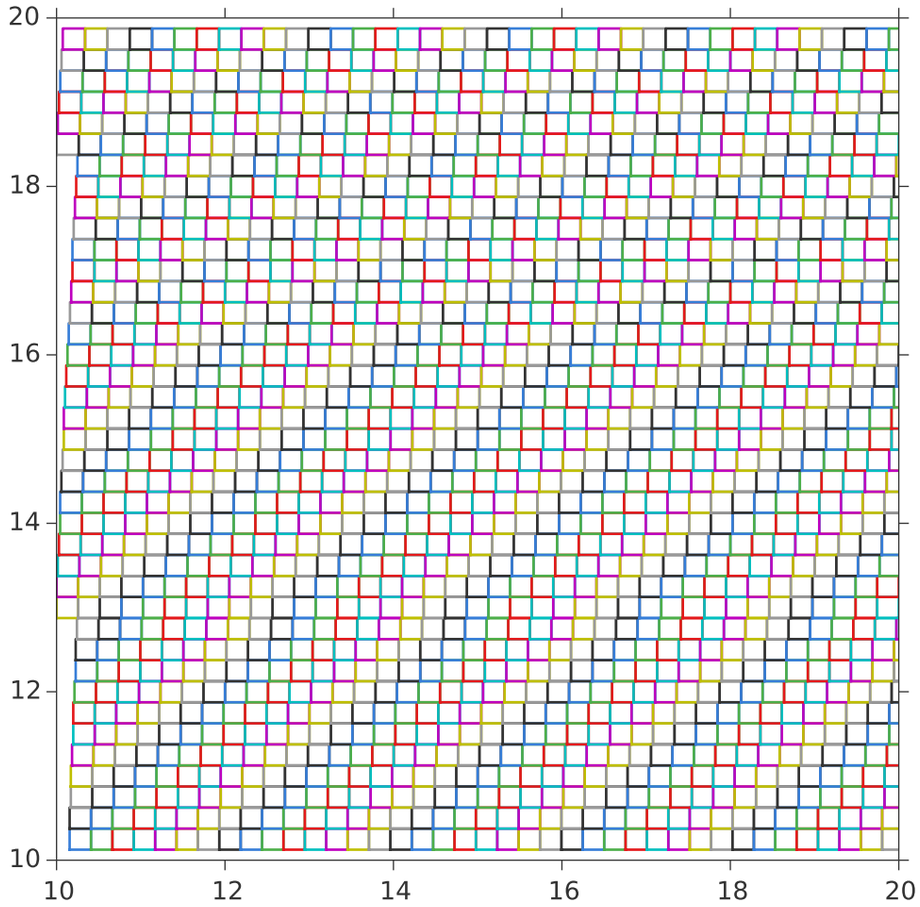
- Design, data formats, etc. are not set in stone

Goals for this workshop (from my perspective)

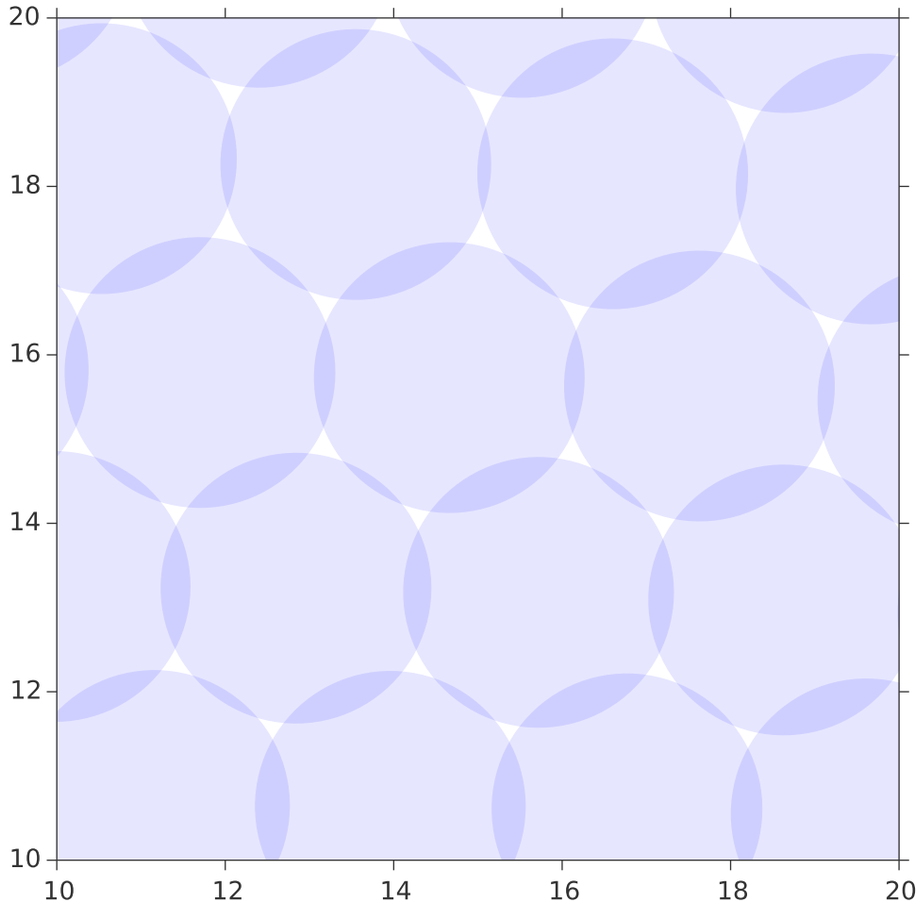
- Each topic understands how their piece fits within the big picture
  — upstream/downstream interfaces understood and match
- Identify missing items and make a plan
- [And make progress on individual pieces]

Imaging	Images –> detected object catalogs
Target Selection	Object catalogs –> wishlist to observe
Fiber Assignment	Wishlist –> actual target:fiber assignments
Operations	Pick pointings, take exposures, write raw data
Spectro Pipeline	Raw data –> useful data (spectra, classifications, redshifts)
LSS Catalog	Redshift catalog + efficiencies (via weights or randoms)

Bricks and Tiles

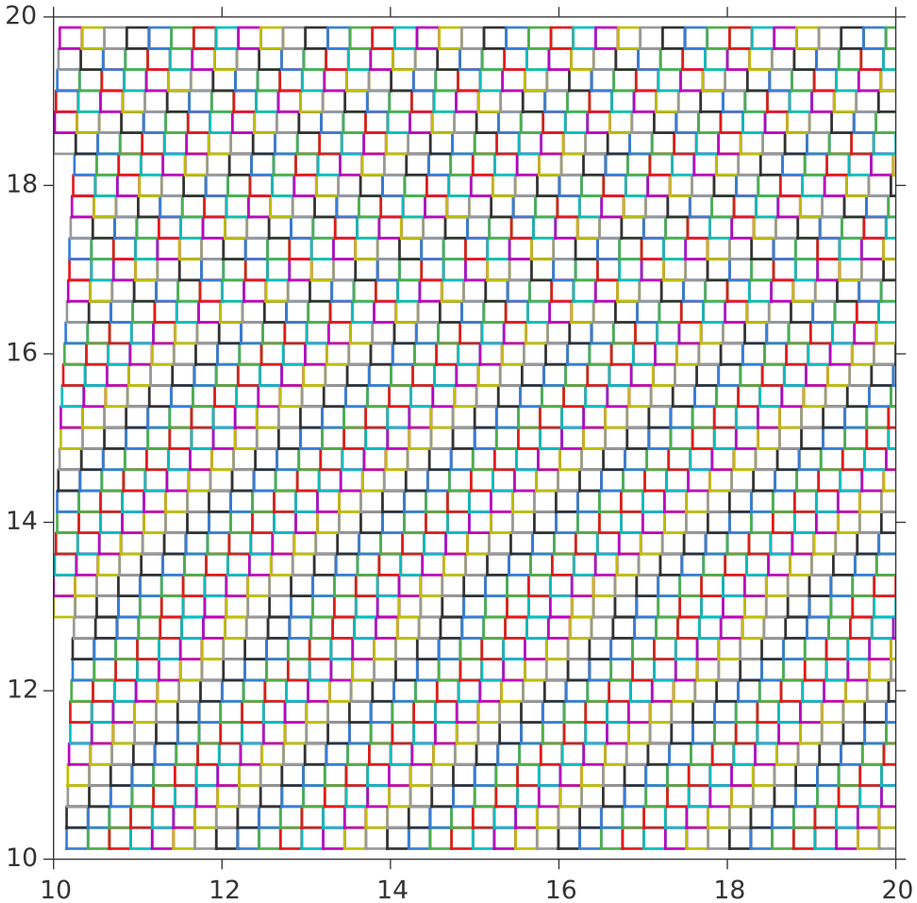


Imaging surveys use “**bricks**”
~0.25 x 0.25 sq deg
edges constant RA or dec

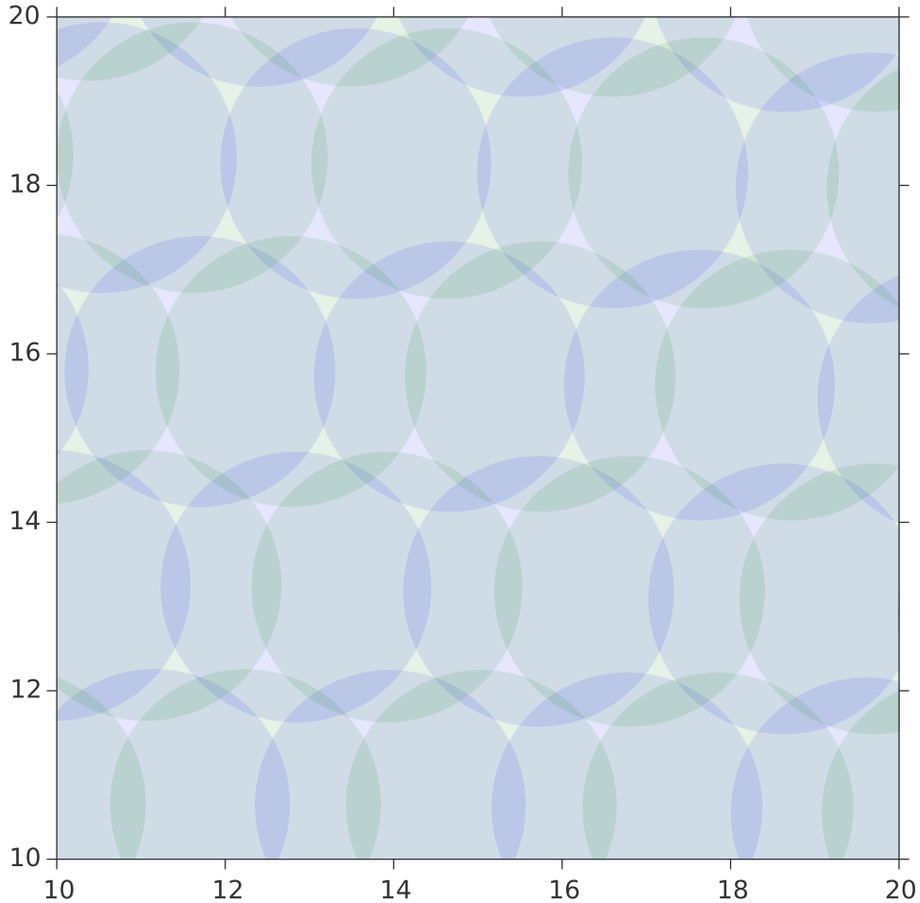


DESI “**tiles**” with 5 overlapping
~8 sq deg pointings

Bricks and Tiles

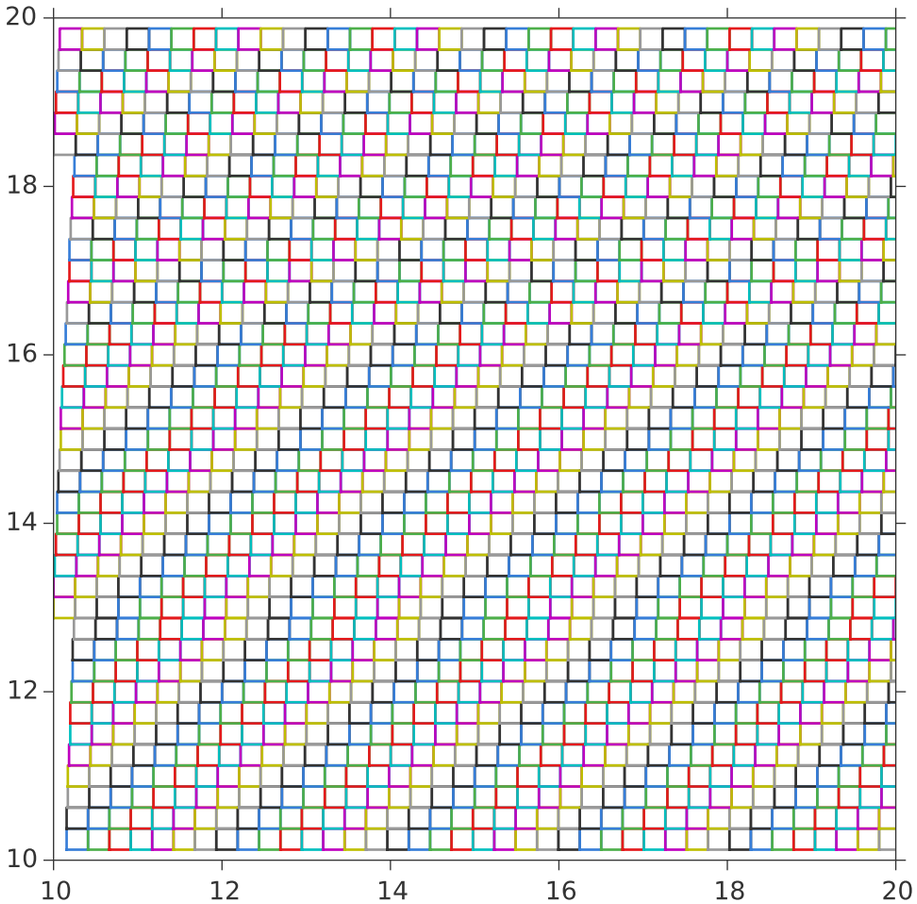


Imaging surveys use “**bricks**”
~0.25 x 0.25 sq deg
edges constant RA or dec

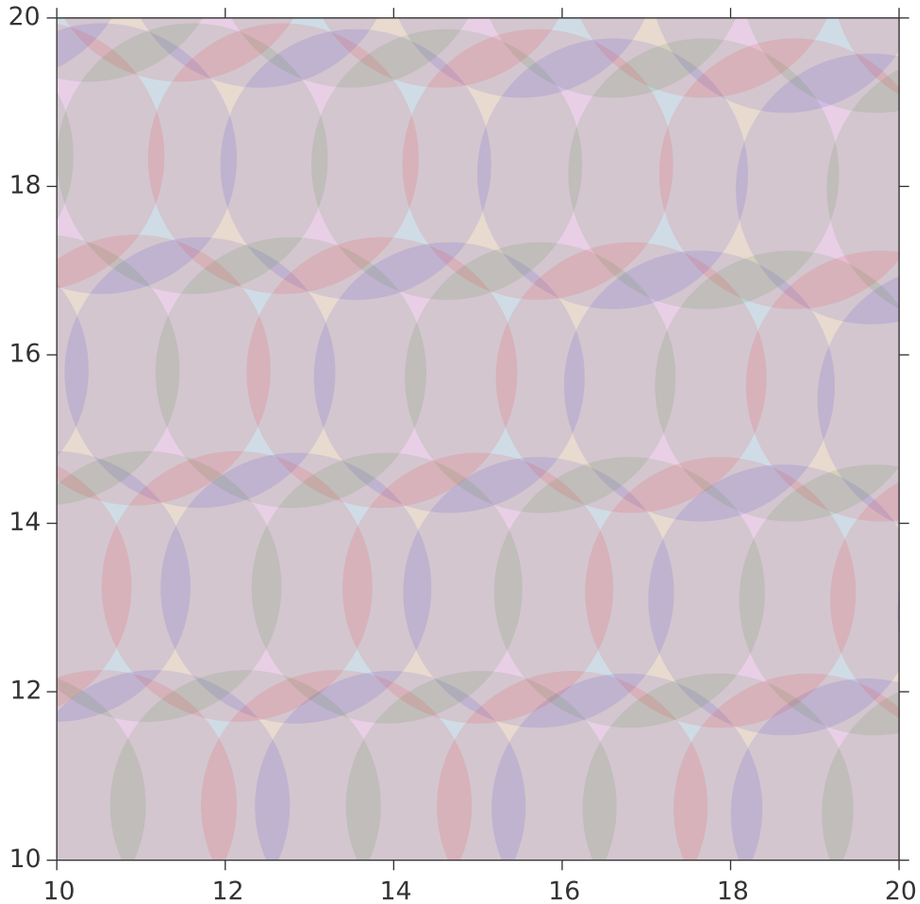


DESI “**tiles**” with 5 overlapping
~8 sq deg pointings

Bricks and Tiles

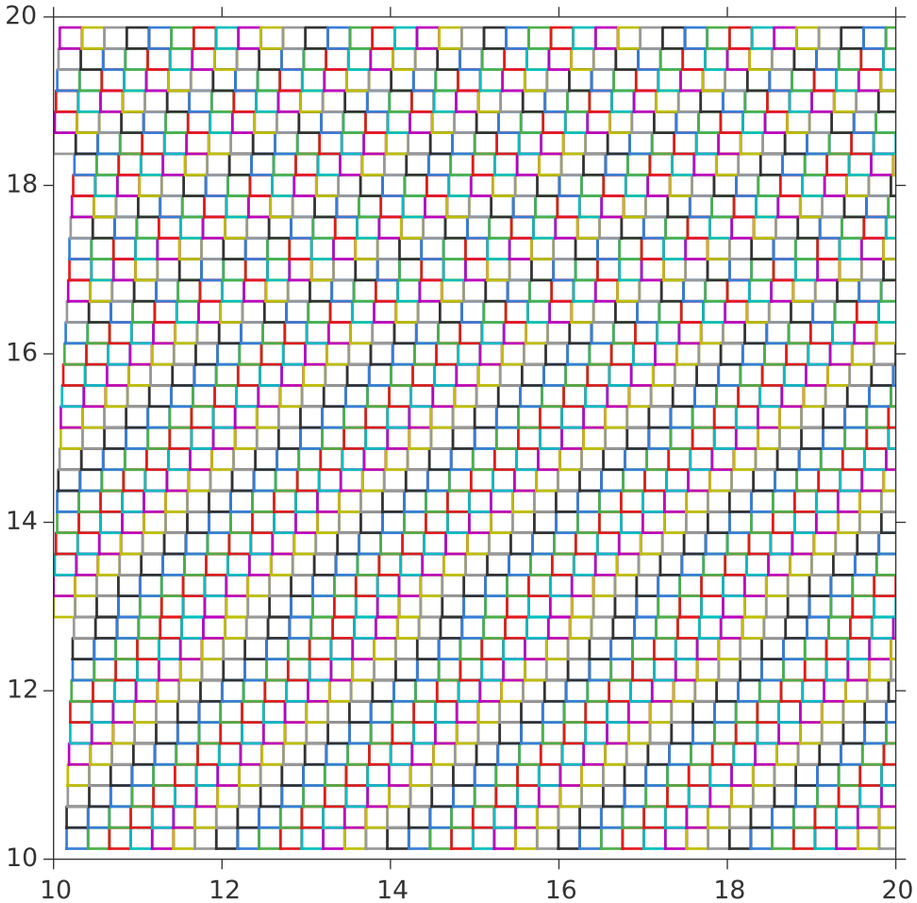


Imaging surveys use “**bricks**”
~0.25 x 0.25 sq deg
edges constant RA or dec

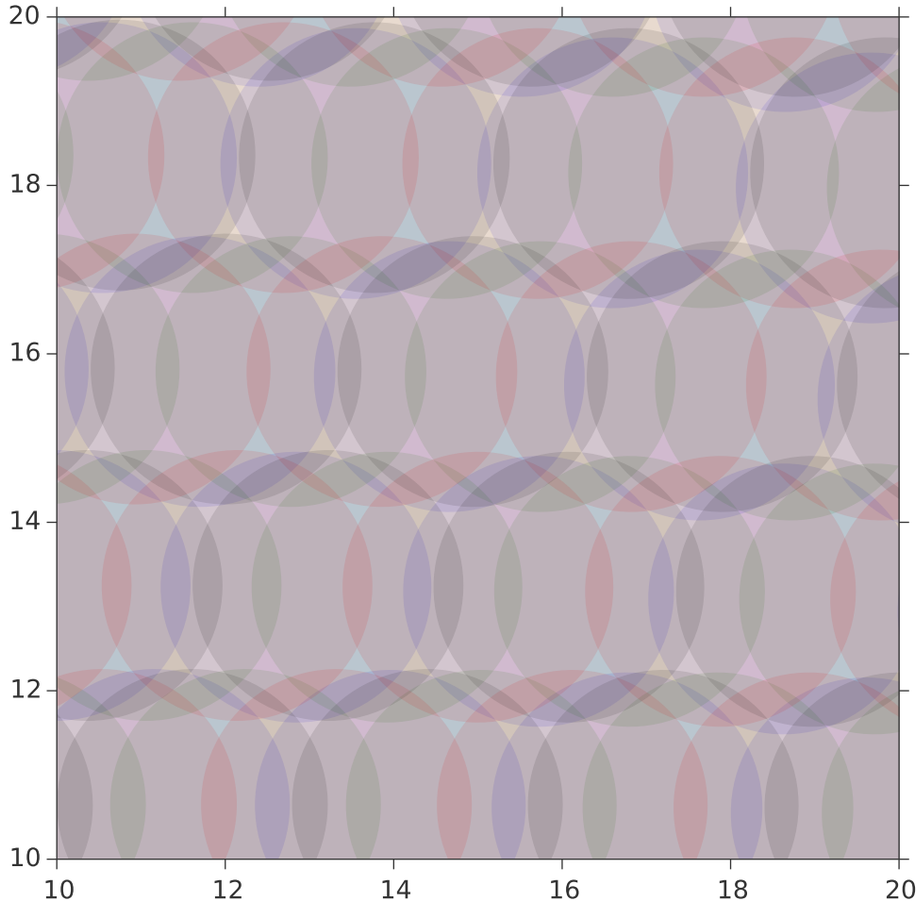


DESI “**tiles**” with 5 overlapping
~8 sq deg pointings

Bricks and Tiles

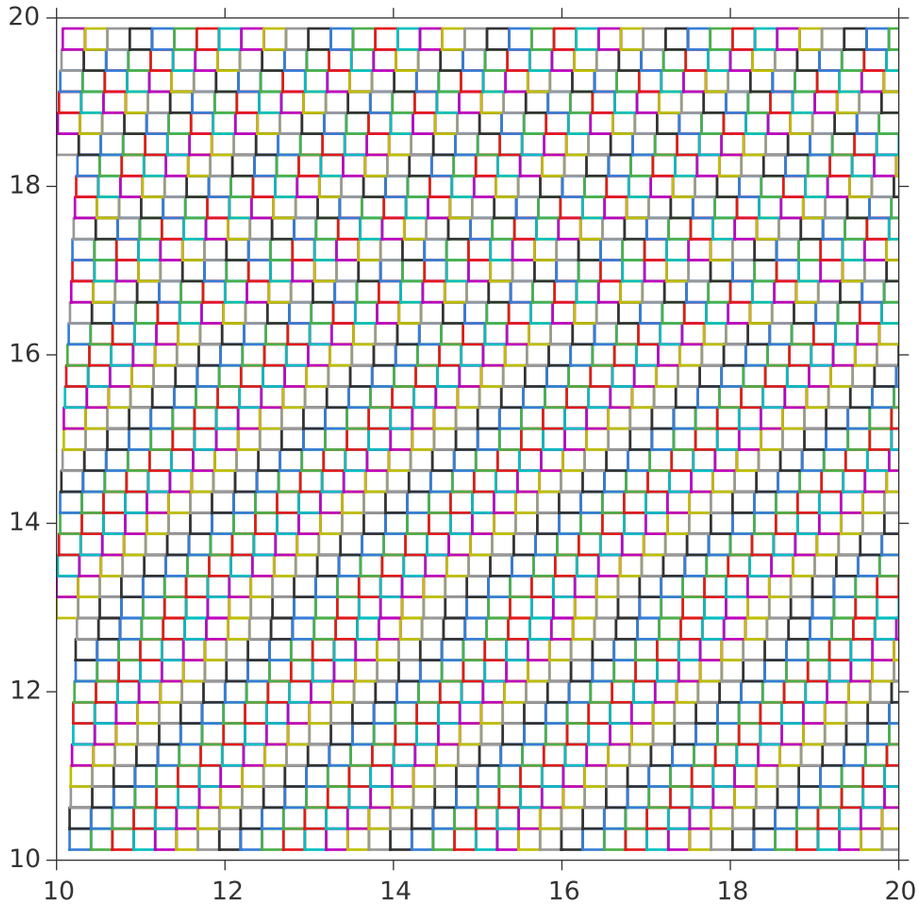


Imaging surveys use “**bricks**”
~0.25 x 0.25 sq deg
edges constant RA or dec

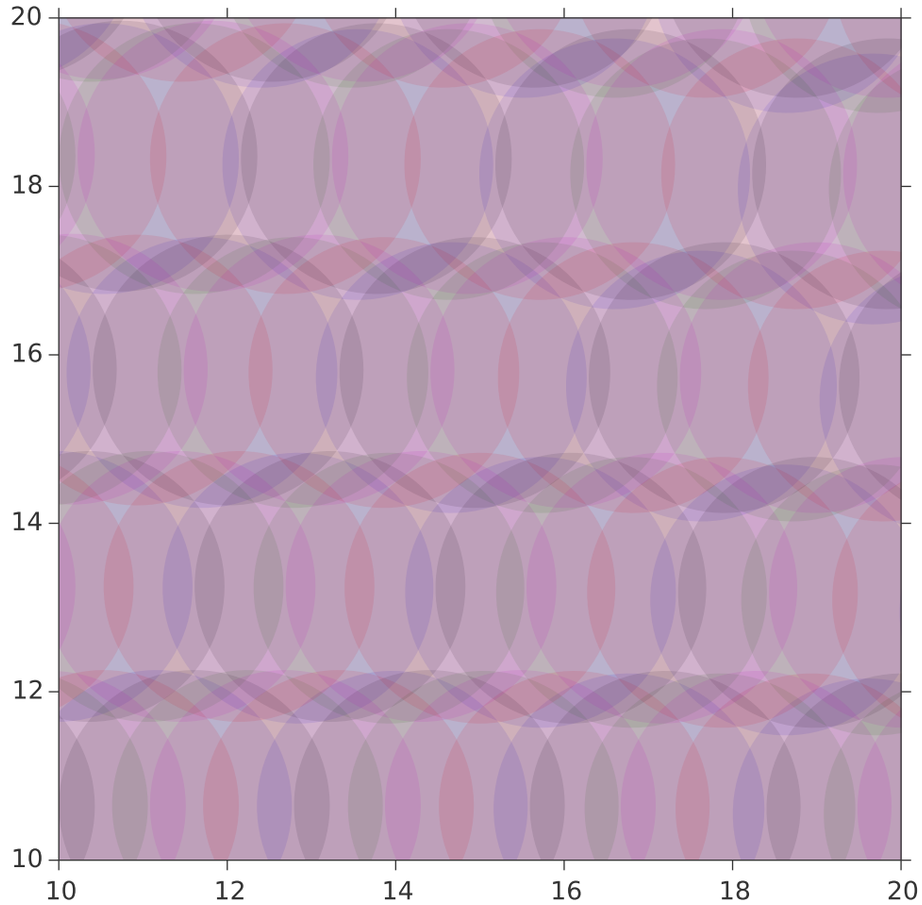


DESI “**tiles**” with 5 overlapping
~8 sq deg pointings

Bricks and Tiles

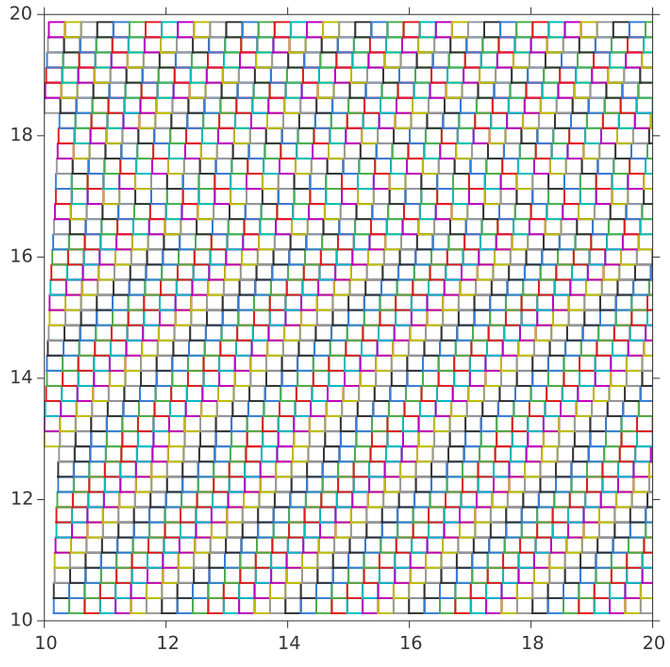


Imaging surveys use “**bricks**”
~0.25 x 0.25 sq deg
edges constant RA or dec



DESI “**tiles**” with 5 overlapping
~8 sq deg pointings

Bricks and Tiles

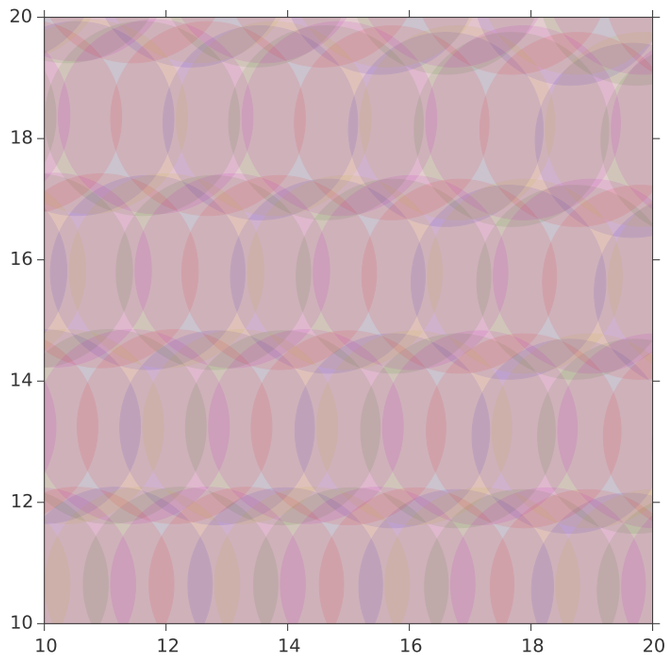


Imaging	Images –> Bricks
Target selection	Bricks –> full footprint
Fiber assignment	Footprint –> tiles
Operations	Tiles –> exposures
Spectro Pipeline	Exposures –> bricks –> footprint
LSS Catalog	Footprint

Different steps use different organizational units,

but they all trace the brickname + targetid

Bricks and Tiles



Imaging	Images –> Bricks
Target selection	Bricks –> full footprint
Fiber assignment	Footprint –> tiles
Operations	Tiles –> exposures
Spectro Pipeline	Exposures –> bricks –> footprint
LSS Catalog	Footprint

Different steps use different organizational units,

but they all trace the brickname + targetid

Imaging

Target Selection

Fiber Assignment

Operations

Spectro Pipeline

LSS Catalog

http://legacysurvey.org

Formally independent of DESI, but closely affiliated

Data from DECam, WISE, Mayall, Bok

"**tractor**" catalogs contain identified objects

***** Do these contain everything we need for LSS tracing?**

"**sweeps**" contain a subset of the tractor catalogs

***** do these contain the subset we need for targeting?**

Organized by "**bricks**" on the sky

— 0.25 x 0.25 sqdeg, iso-RA, iso-DEC boundaries

— used for downstream object grouping

— exact size under negotiation

***** How much information do we propagate forward, vs. just propagating object IDs?**

Imaging

Target Selection

Fiber Assignment

Operations

Spectro Pipeline

LSS Catalog

https://github.com/desihub/desitarget

Spins over tractor catalogs, makes cuts,
writes a single output file (i.e. not by brick)

Target **bitmasks** track which targets pass which cuts
— allows same object to pass multiple cuts

Output quantities

http://desidatamodel.readthedocs.org/en/latest/DESI_TARGET/targets.html

— variables used for target selection
e.g. fluxes
— variables needed for fiber assignment
e.g. RA, dec
— variables needed for traceability
e.g. brickname, targetid, targetflag

***** Is this sufficient? Is this convenient?**

***** Is database version needed?**

Imaging

Target Selection

Fiber Assignment

Operations

Spectro Pipeline

LSS Catalog

https://github.com/desihub/fiberassign
Data flow is an active work in progress

"**Merged Target List**" combines multiple catalogs and feedback from spectro pipeline, outputs organized minimal information needed by fiber assignment itself
— RA, dec, priority + brickname, targetid, targetflag

FA writes **one file per tile**; for each tile/fiber:
— what target was assigned
— what targets could have been assigned?
— http://desidatamodel.readthedocs.org/en/latest/DESI_TARGET/fiberassign/tile.html

Missing
— target-oriented view (vs. tile-oriented)
— why an object was (not) picked
— probability that a target would be picked

***** LSS catalog people: scrutinize this**

Imaging

Target Selection

Fiber Assignment

Operations

Spectro Pipeline

LSS Catalog

Things to **pay attention** to

— tile priorities impact data sets per year

— dynamic exposure time calculation affects uniformity of depth/efficiency

Feedback loop from spectral pipeline results impact future fiber assignments & operations

— e.g. does this QSO target need more exposures?

— need to simulate this to realistically assess survey performance and LSS catalog weights

Outputs

— spectral raw data

— guide camera info

— fibermap (as-implemented version of fiber assignment)

***** LSS catalog: what do you need from survey?**

Imaging

Target Selection

Fiber Assignment

Operations

Spectro Pipeline

LSS Catalog

Raw data –> **spectra, classifications, redshifts**

Early steps organized by **exposure**;
most users won't need this level

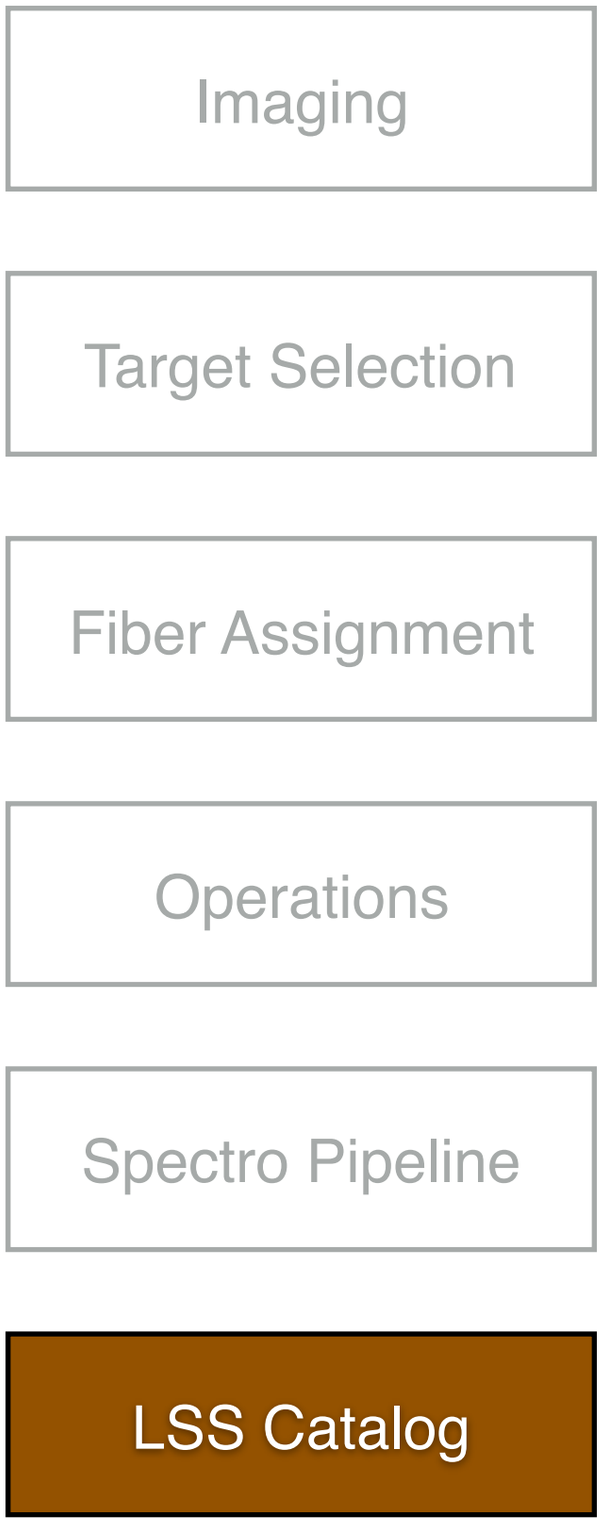
Later steps regrouped by **brick**

http://desidatamodel.readthedocs.org/en/latest/DESI_SPECTRO_REDUX/PRODNAME/index.html

— same grouping as tractor bricks
— individual exposures, cameras (brick*.fits)
— coadds across exposures, across cameras
(coadd*.fits; currently not included)
— classifications and redshifts (zbest*.fits)

zcatalog: regroups all zbest files back into one file

Missing:
— how to express pipeline efficiency?
— merge of results with targets that were never observed
— targetid –> brick



No code integrated with the rest of the system yet

Open questions (AFAIK):

— track efficiencies via weights or randoms or both?

— how to extract efficiencies from
  * imaging
  * target selection
  * fiber assignment
  * operations
  * spectro pipeline

— data model to express results?

Multiple Surveys

How closely coupled should dark/BGS/MWS/other be?

- e.g. do targetids need to be unique across surveys?
  - non-trivial since they come in from different sources
- Does processing output need to be merged, separated, don't care?
  - If separated, who gets miscellaneous ancillary targets (SN hosts, etc.)

Available Simulators

Specsim

- Input spectrum –> output spectrum
  - Throughput, resolution, statistical noise
- http://github.com/desihub/specsim (lead: David Kirkby)
- Refactored from original “quicksim” in desimodel

Pixsim / Specter

- Input spectrum –> CCD pixels
- http://github.com/desihub/specter + desisim (lead: Stephen Bailey)
- script: desisim/bin/pixsim-desi

Quickgen

- Wraps original quicksim –> output DESI pipeline format files
- http://github.com/desihub/desisim
- script: desisim/bin/quickgen
- lead: Govinda Dhungana

Simulators to do

Quickcat

- renamed from “quickz”
- Fiber assignment output –> spectro pipeline catalog (or LSS catalog?)
- v0beta exists: redshifts in = redshifts out

Opsim

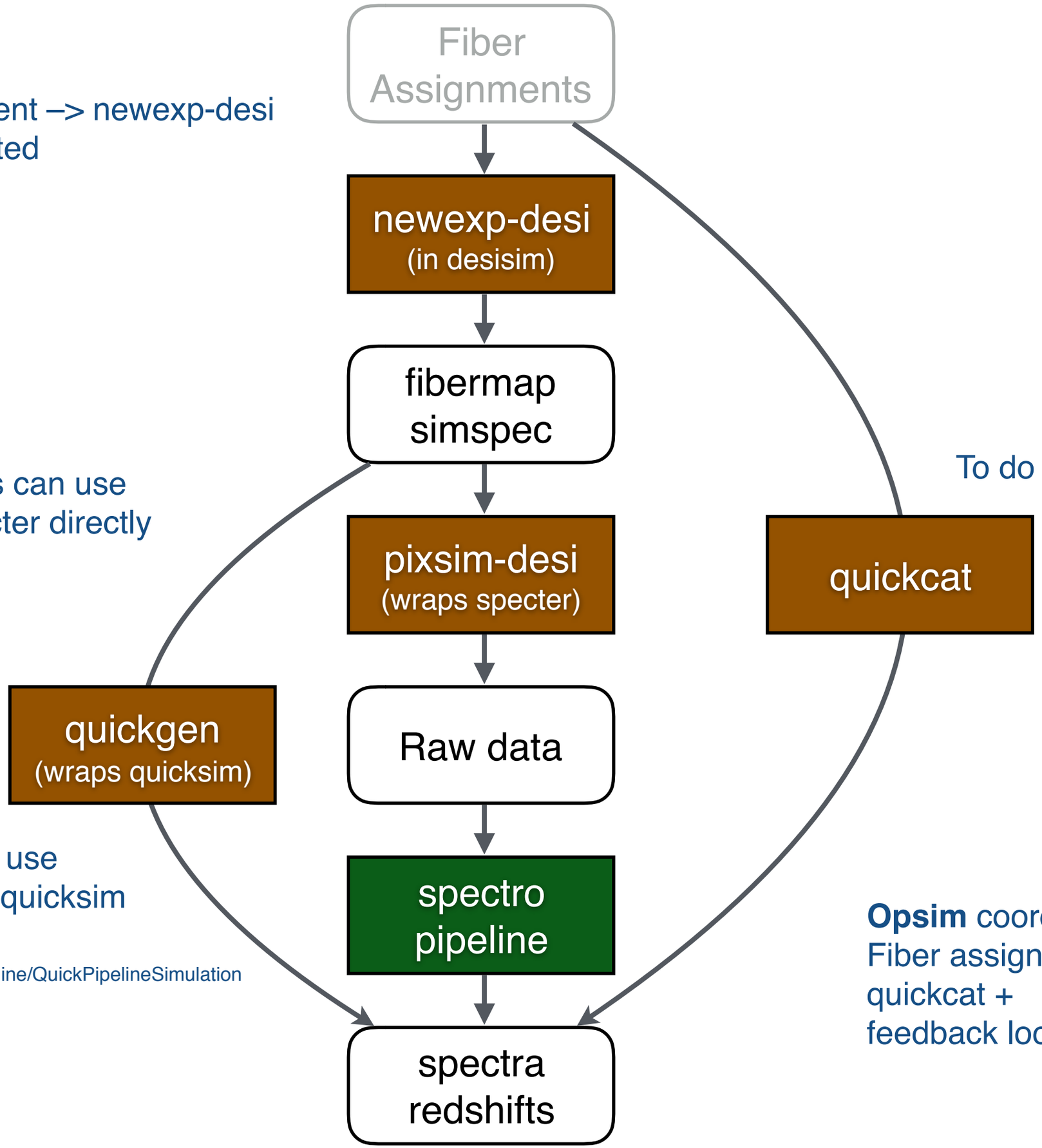
- Operations feedback loop
  - fiber assignment –> observations –> spectro pipeline –> update fiber assignment
  - repeat
- v0: plan out in big chunks
- v1: integrate with next field selector and weather model to go tile by tile

Fiber assignment –> newexp-desi
not yet integrated

Focused studies can use
specsim or specter directly

Soon will update to use
specsim instead of quicksim

https://desi.lbl.gov/trac/wiki/Pipeline/QuickPipelineSimulation



To do at this workshop

Opsim coordinates
Fiber assignment +
quickcat +
feedback loop

Data Challenges

Focused efforts on specific topics to organize work

- DC1: spectro algorithms applied to BOSS data
- DC2: spectro algorithms applied to DESI pixel simulations
- DC3 (delayed): automate spectral pipeline
- DCn (current): Fiber assignment –> ops –> spectro pipeline feedback loop
- DCn+1 : what does your group need?
  - — We need to simulate stuff for development anyway; it might as well be useful to you too
  - — We won't do your science data challenge for you, but we are happy to help with tools

Upcoming

- Spring 2016: process teststand data (first real DESI spectro data!)
- Fall 2016: open, but likely end-to-end / scaling tests, in prep for
- 2017: Full (dark) survey end-to-end pixel-level
  - — This will be as detailed & useful as you make it
- 2018: fix what we learned from 2017

A bunch of links

Data at NERSC

Web access	https://portal.nersc.gov/project/desi/collab/spectro/redux
Tractor files	/project/projectdirs/cosmo/data/legacysurvey/dr1/tractor/
Target selection	/project/projectdirs/desi/target/targets-dr1-test.fits
Fiber assignment	/project/projectdirs/desi/target/fiberassign/durham1-0.0/
Simulated raw data	/project/projectdirs/desi/spectro/sim/cosmics_test/
Spectro pipeline	/project/projectdirs/desi/spectro/redux/elm/

***Caveat**: these don't yet chain together (maybe by the end of this week they will!)*

Reference pages

https://desi.lbl.gov/trac/wiki/Computing/DataFlowIntro (wiki version of this talk)
http://legacysurvey.org/dr1/catalogs/ (tractor file data model)
https://desidatamodel.readthedocs.org
https://desi.lbl.gov/trac/wiki/Pipeline/QuickPipelineSimulation
https://desi.lbl.gov/trac/wiki/Pipeline/QuickSim

Working within the DESI software eco-system

https://desi.lbl.gov/trac/wiki/Computing/AccessNersc (How to get a NERSC account)
https://desi.lbl.gov/trac/wiki/Computing/Software/Using (DESI environment at NERSC)
https://desi.lbl.gov/trac/wiki/Computing/Software/Guidelines
https://desi.lbl.gov/trac/wiki/Computing/UsingGit