Serverless

Up-to-date Infrastructure Diagram

AWS

Amazon S3
Our application is deployed and stored here -- this includes all assets.

Amazon CloudFront
We'll used CloudFront as a distrubution tool to deliver our static assets from S3

user

Amazon API Gateway
The static application will access all dynamic content and actions by hitting endpoints available in the API Gateway

Backround Jobs
This will replace DelayedJob classes and will be initiated by CloudWatch Events

Amazon Lambda

CMR
The static application won't make any calls to CMR directly, we'll wrap them in Lambda functions

User Requests
These functions will be only be called from API Gateway with requests originating from the UI

Amazon CloudWatch
We'll use CloudWatch for our logging instead of Splunk. We'll also use CloudWatch Events and Lambda for any cron jobs that remain.

Amazon RDS
Our datastore will remain here unless we can make a case for moving to Dynamo which I can't see happening straight away

Amazon SQS
We will plan to remove DelayedJob and instead take advantage of the non-database queue system

AWS cloud

Database Entity Relationship Diagram

projects

id
name
path
user_id

users

id
echo_id
echo_profile
echo_preferences
urs_id
urs_profile
site_preferences

retrievals

id
user_id
jsondata
token
environment

colormaps

id
product
url
jsondata

shapefiles

id
user_id
file
file_hash

access_configurations

id
user_id
collection_id
service_options
echoform_digest

retrieval_collections

id
retrieval_id
access_method
collection_id
collection_metadata
granule_params
granule_count

orders

id
retrieval_collection_id
type
search_params
order_number
state
order_information

Submitting an Order

Submit Order →

Queues Delayed Job

Processing the Order

Search for collection

Stage for Delivery

Customize

Stage for Delivery

:Application | :Legacy Services | :CMR | :URS

Get Graunles

Get Order Information

Create Empty Order

Add Granules to Order

Get User Preferenecs

If no preferences exist for the user

Add Preferences

If the user now has preferences but lacks valid contact information

Get ECHO User

Get URS User

Update Preferences

Get ECHO User

Add User Information
to the Order

Submit Order

Customize

:Application	:Legacy Services	:CMR	:URS

Get Service URL

Get Option Assignment
Information

Get Service Entry

Get Graunles

*Submits to the service url determined
by the get service url above*

Submit Order