ChimeraX LigandRecognizer Tutorial

This document provides a step-by-step tutorial on how to use the **`blob`** commands from the ChimeraX LigandRecognizer bundle. **`Blob`** commands implement the functionality proposed in "*Ligand Identification in CryoEM and X-ray Maps Using Deep Learning*" which uses a deep learning model to predict ligand types. **`Blob`** commands allow validation of previously modeled ligands or tentative identification of unmodeled or incorrectly identified ligands.

The **`blob`** commands assume that both a .pdb/.cif model (possibly partial or ligand-free) and a .ccp4 map are loaded into ChimeraX. A key requirement is that the map is a map-versus-model *difference map* in which high density suggests unmodeled ligands rather than the experimentally determined cryoEM or X-ray map. Difference map construction is described below.

Installation

Download and install the latest version of the LigandReconizer bundle from:
https://github.com/wtaisner/chimerax-ligand-recognizer.
In the future, this bundle will also be available through the ChimeraX Toolshed.

After installation, open ChimeraX and type in the ChimeraX Command Line:
`help blob`
If installation was successful the blob command help page will open in the ChimeraX browser.

To construct difference maps, you will also need Phenix, which can be installed from https://phenix-online.org/download/.

Download or prepare tutorial data

In this tutorial, we will use PDB entry 8FUZ as an example. BFUZ is an octameric Inosine-5'-monophosphate dehydrogenase 2; each of the 8 subunits contains NAD and nearby IMP ligands which have already been modeled.

You can download:

- the full 8FUZ model (8FUZ.cif)
- the partial model (8FUZ_stripped.cif) with ligands removed
- difference map (map_model_difference_1.ccp4)

from: https://1drv.ms/f/s!Aq419F62GZU4g_tDT9LFuNWVV9Eohg?e=8vUR7W. Subsequent instructions assume these files are in a subdirectory, 8FUZ/ of the ChimeraX working directory. If you choose a different location modify paths in tutorial commands accordingly.

Alternatively, to construct your own partial model and difference map:

- Download 8FUZ.cif and the corresponding map, emdb_29482.map, either using ChimeraX or directly from PDB and EMDB.
- Create a partial (ligand-free) model, *e.g.* via
  `grep -v HETATM 8FUZ.cif >8FUZ_stripped.cif`
  in a Linux environment or with a text editor.
- Generate the difference map either by running **`computeMapModelDifference.sh 8FUZ emd_29482.map 8FUZ_stripped.cif`** or by directly running the Phenix command **`phenix.real_space_diff_map 8FUZ_stripped.cif emd_29482.map Resolution=2.1`**. (Resolution, here 2.1 Angstroms, is taken from the model). Script **`computeMapModelDifference.sh`** is included in the blob tool repository.

Commands

The **`blob`** tool implements three basic commands:

1. **`blob autothreshold [map_id] [style] [density_std_threshold]`**
   Automatically choose or manually set a map threshold and optionally adjust the display format.
2. **`blob recognize [map_id] [surface_id] [pdb_id] [flg_xray] [density_threshold]`**
   Recognize (predict identity of) a map fragment bounded by **`surface_id`** in **`map_id`**.
3. **`blob validate res_id [map_id] [pdb_id] [flg_xray] [density_threshold]`**
   Validate (prediction identity) of a previously modeled ligand at **`res_id`**.

Parameters in brackets are optional, and, if omitted, the command will use the currently active map, structure/model, and surface and will assume that the map is a cryoEM map (flg_xray = 0). Keyword argument identifiers may also be used and are defined in the **`help blob`** page. The above three commands are aliases for `blobus autothresholdus, blobus recognitus, and blobus validatus`, and respectively.

The following sections show step-by-step examples of how to use these commands in practice.

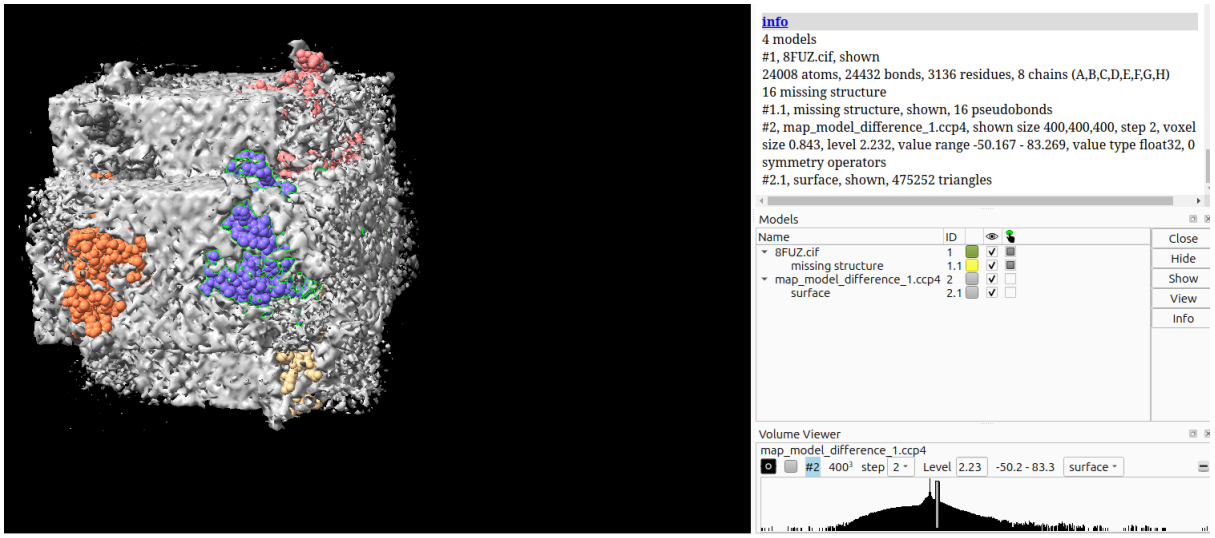
Blob validate: checking a previously modeled ligand

1. Open the full model and the difference map

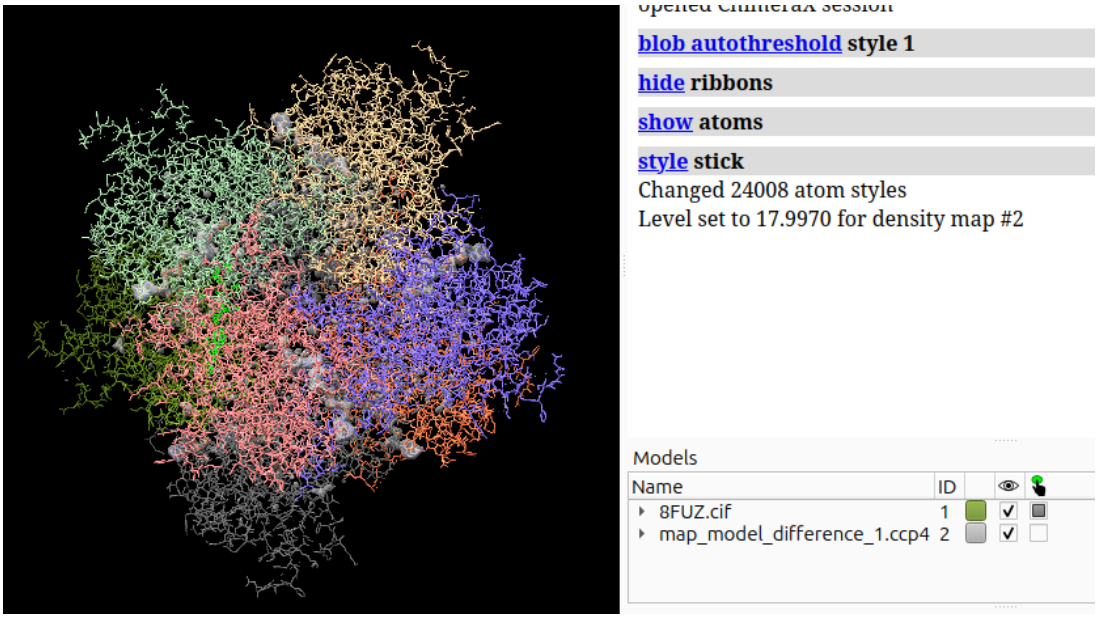
   `ChimeraX 8FUZ/8FUZ.cif 8FUZ/map_model_difference_1.ccp4`

Using a fresh ChimeraX session is recommended so we can predict the IDs of the various components. *E.g.* here the full model and difference map will be #1 and #2, respectively. If you choose to work in an existing ChimeraX session instead, adjust the ID numbers used throughout the tutorial accordingly.

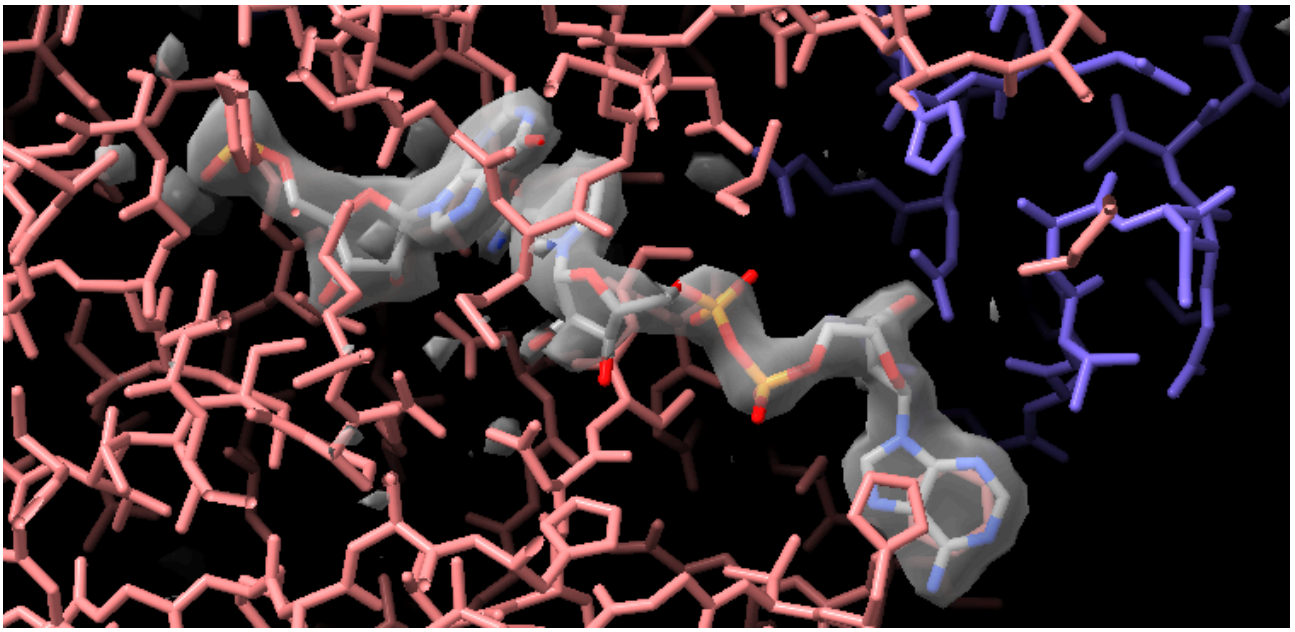
The model and map will open, but ChimeraX's default threshold and display settings yield poor visibility.



2. Adjust the view
   a. Run the ChimeraX command **`blob autothreshold style 1`** to automatically choose a threshold, make surfaces partially transparent, and display atoms as sticks. (Style 2 would display backbone ribbons, while 0 would only threshold without changing style).
   b.



c. Run **`view /B:601`** to see the part of the density map we'll be working on:



3. Validate the selected ligand:

a. To validate residue with id 601 in chain B, run: **`blob validate /B:601.`** Equivalently, you can specify the residue, map, and model manually, *e.g.*:
`blob validate /B:601 #2 #1`

b. After a short while, in the ChimeraX log, you should see something similar to the following:

blob validate /B:601
Attempting to cut ligand (this may take a while)...
23 atoms, 25 bonds, 1 residue, 1 model selected
Ligand Class Predictions
(Click on ligand group name to see the full list of ligands)

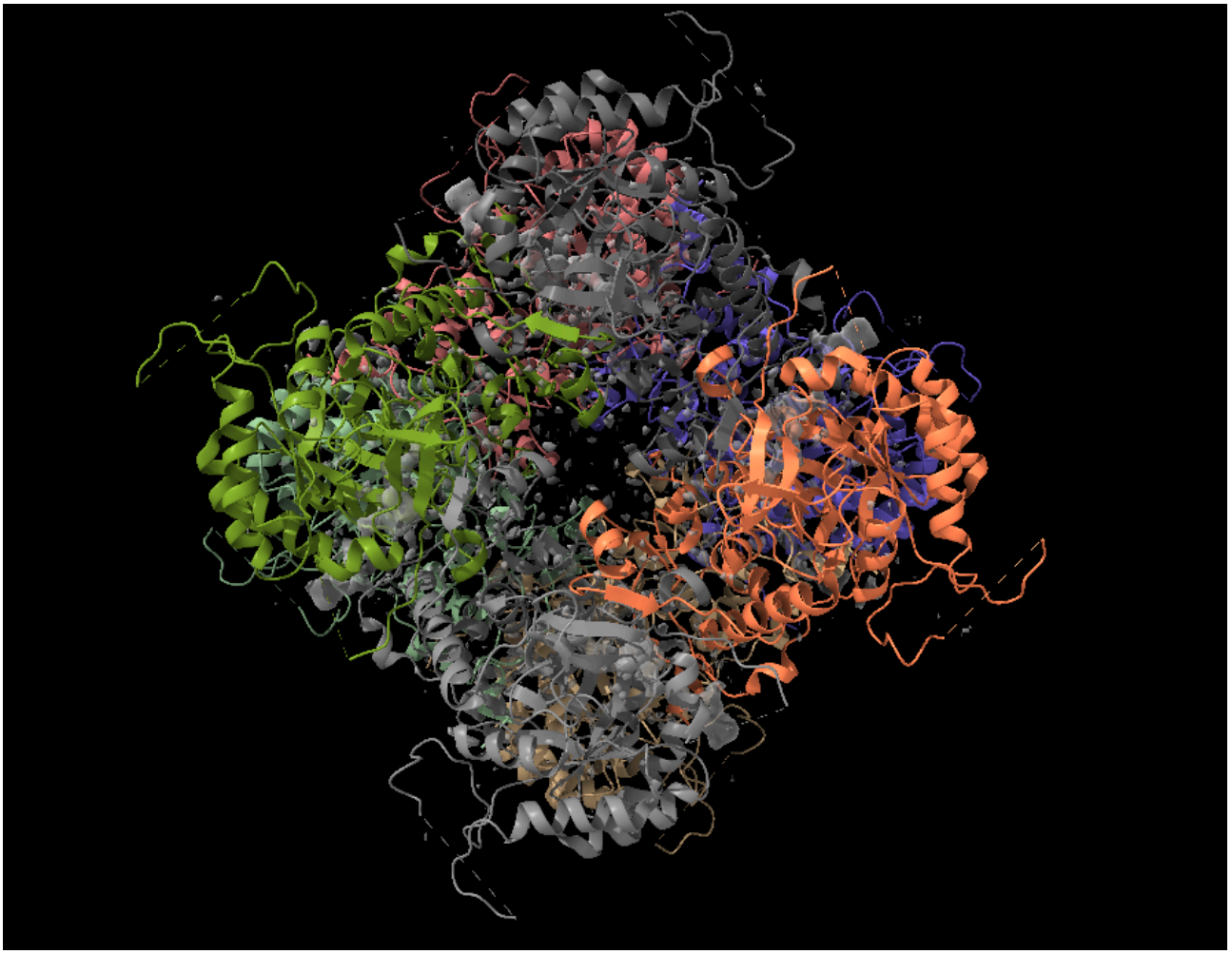
Ligand Class	Probability
AMP-like	0.999
ADP-like	0.000
RARE_LIGAND	0.000
5GP-like	0.000
ATP-like	0.000
GDP-like	0.000
AGS	0.000
HEA	0.000
FMN-like	0.000
A3P	0.000

c. The results indicate with very high confidence (.999) that the ligand is AMP-like. Left-click on AMP-like to view a list of specific ligands in this class. Hovering over a member of the resulting list will display the structure, while left-clicking will open the corresponding PDB ligand page. Note that inosinic acid (IMP) is a member of AMP-like, in agreement with the modeled identity shown by **`info residue /B:601:`**

info residues /B:601
residue id /B:601 name IMP

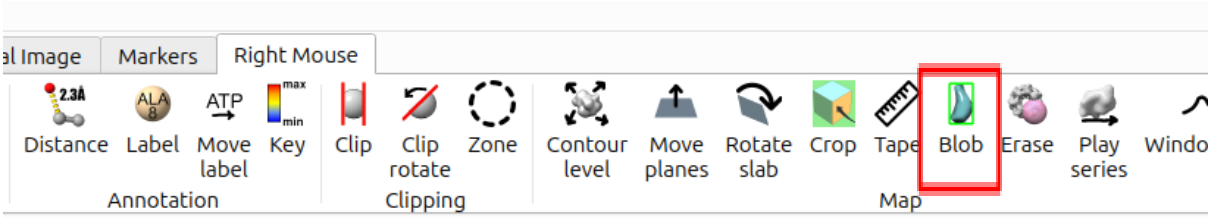
Blob recognize: predict the identity of unknown ligand(s)

1. Open files, autothreshold, and set display options as before
   a. Run: `ChimeraX 8FUZ/8FUZ_stripped.cif 8FUZ/map_model_difference_1.ccp4`
   b. Run the ChimeraX command `blob autothreshold style 2`:

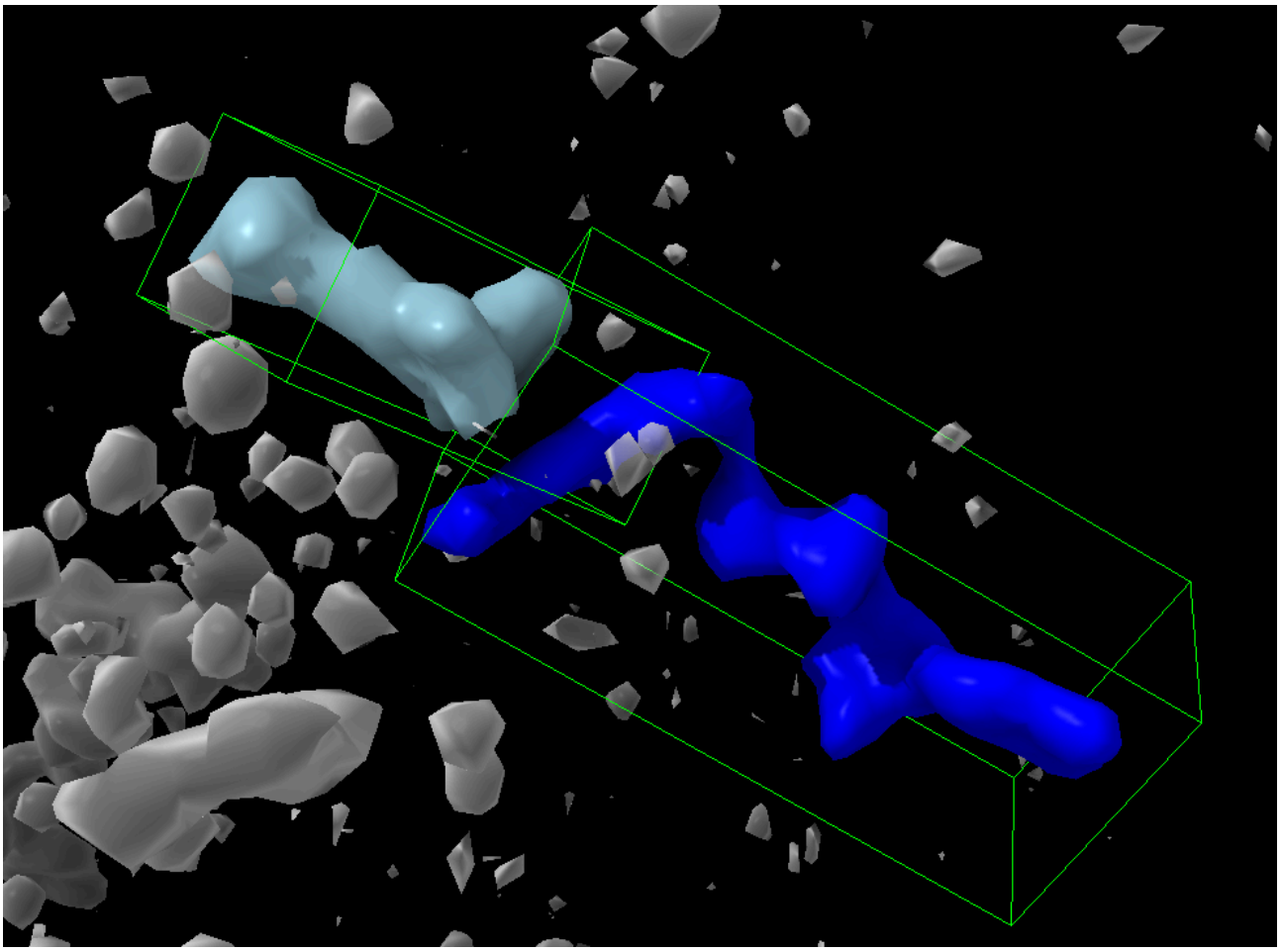


We have again used a fresh ChimeraX session to generate known IDs for the tutorial. The partial, rather than full, model has been chosen to emulate a situation in which ligands are unknown and have not yet been modeled.

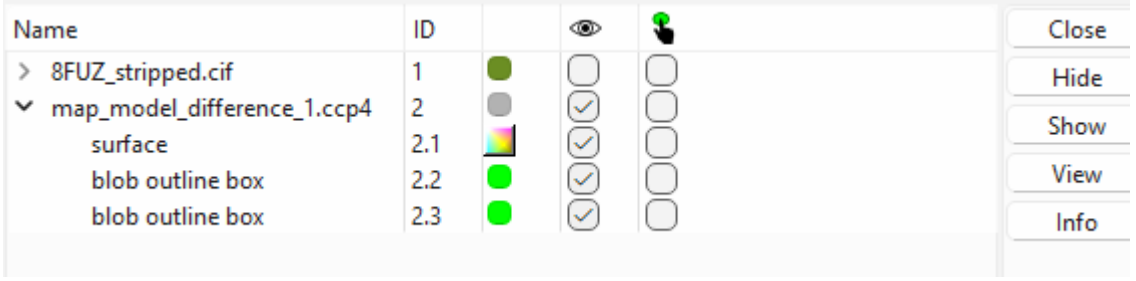
2. Select blob(s) for identification
   a. Run the ChimeraX command `hide atoms ribbons` to hide the model, leaving only the difference map visible. (Other workflows leaving part or all of the model visible are also possible, and will be outlined below).
   b. From the ChimeraX menu, select Right Mouse / Blob, enabling this tool



c. Using left mouse and drag to rotate, middle mouse and drag to translate, and the mouse wheel to zoom, focus on an area of interest, and then right-click on blobs of interest. In the example below, we have selected a pair of neighboring ligands of different types, which assigns each of them a unique color.

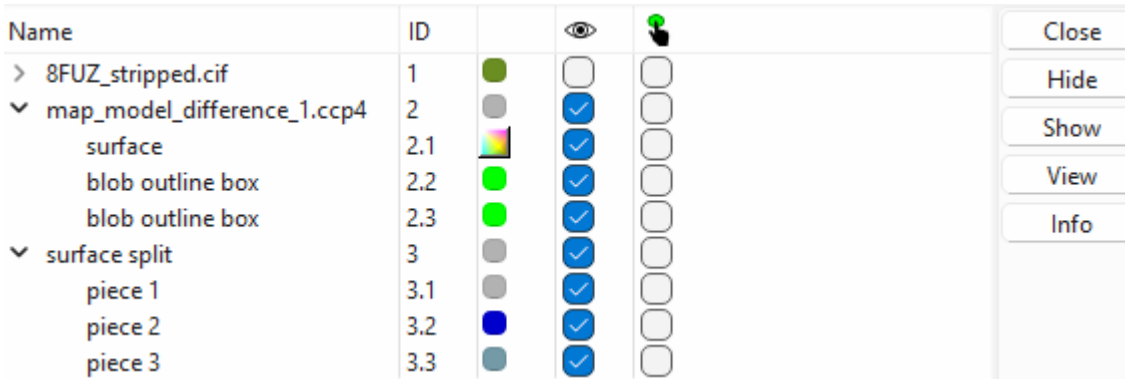


Although they have been colored and boxed, the selected blobs remain part of the original map whose surface includes the entire thresholded volume:



As a result, they are not yet suitable for use with **`blob recognize`**.

d. **`Blob recognize`** requires each blob of interest to be delimited by its own, unique surface. Generate suitable surfaces via **`surface splitbycolor #2.1`**, yielding:



Here #3.2 and #3.3 are surfaces bounding the 2 selected blobs (denoted by their color), while # 3.1 contains the remaining, gray blobs.

3. Predict the blob identities
   a. Run **`blob recognize #2 #3.2`** and note that this blob is correctly identified as NAD-like.
   b. Run **`blob recognize #2 #3.3`** and note that this blob is also correctly identified as AMP-like.

The workflow just described is particularly effective when you are interested in identifying many or all unidentified blobs in the difference maps without regard to their position in the model. In other cases, you may be interested in particular regions and may wish to leave the model visible or partly so to facilitate identification of these regions. Several alternative approaches are available and can be combined as desired. Below, we note some of the available tools.

Clipping planes can be used to limit both model and map visibility, preventing overlying portions of either the model or map from blocking your view. See **`help clip`** for details. Similarly, enabling the Outline Box (item Map / Outline Box) and the Right Mouse / Crop tool allows you to select and drag the faces of the Outline box, cropping out a subset of the map. Cropping affects only the map and not the model.

The Erase tool (Right Mouse / Erase) displays an eraser sphere (pink by default) which you can size and position in 3D (click and drag with the right mouse button). Once positioned as desired, you can choose to erase everything inside or outside of the sphere. The first erasure creates a new, partially erased map, leaving the original unaltered; subsequent erasures on the new map do not result in further duplication.

By repeating and perhaps combining these operations, you can select blob(s) of interest with high precision. Finally, keep in mind that blob recognition and validation is affected by the chosen, surface-defining threshold; while the automatically chosen threshold often works well, this is not always the case.