



STRUCTURAL BIOLOGY
COMMUNICATIONS

Volume 74 (2018)

Supporting information for article:

***Cinder*: keeping crystallographers app-y**

Nicholas Rosa, Marko Ristic, Bevan Marshall and Janet Newman

1. Login

Authenticate the user with the system

Request

Method	URL
GET	host/api/login/getauthenticated

Type	Params	Values
HEAD	Authorization	Basic Base64("Username:Password")

Response

Status	Response
200	Content-Type: application/json true
401	{"Message": "Authorization has been denied for this request."}

2. Get image ids

Gets a list of unique image ids that should be shown to the user

Request

Method	URL
GET	host/api/images/getuserimages/

Type	Params	Values
HEAD	Authorization	Basic Base64("Username:Password")

Response

Status	Response
200	Content-Type: application/json An array containing the ID's of images to be shown, ids are referred to as filenames but can be any unique identifier Example response:- [{"filename": "1"}, {"filename": "2"}, {"filename": "3"}]
401	{"Message": "Authorization has been denied for this request."}

3. Get Image

Get more information on a particular recipe

Request

Method	URL
GET	host/api/images/getimagestream/{imageId}

Type	Params	Values
HEAD	Authorization	Basic Base64("Username:Password")
URL	imageId	string

imageId

Id of the image you want the image file of.

Response

Status	Response
200	Content-Type: image/jpeg
401	{"Message": "Authorization has been denied for this request."}

4. Save scores

Publishes a score to the database

Request

Method	URL
GET	host/api/score/publishscore/{imageId}/{score}

Type	Params	Values
URL	imageId	string
URL	score	['X', 'C', 'P', 'O']
HEAD	Authorization	Basic Base64("Username:Password")

imageId

The unique identifier of the image being scored.

score

The score to publish.

'X' = Crystal

'C' = Clear

'P' = Precipitate

'O' = Other

Response

Status	Response
200	Content-Type: application/json bool
401	{"Message": "Authorization has been denied for this request."}

Returns true for success and false for failure.

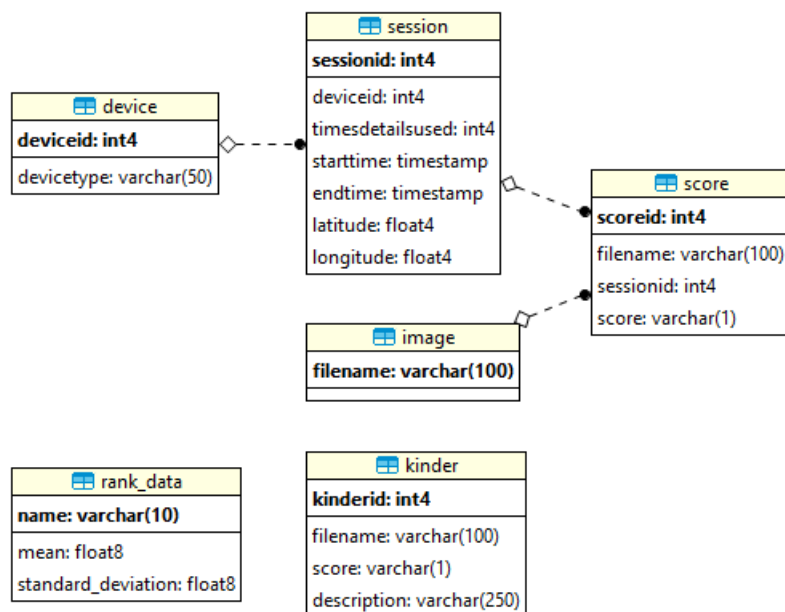


Figure S1 The table structure for the PostgreSQL database which supports the Cinder Kinder and Cinder Community functions.

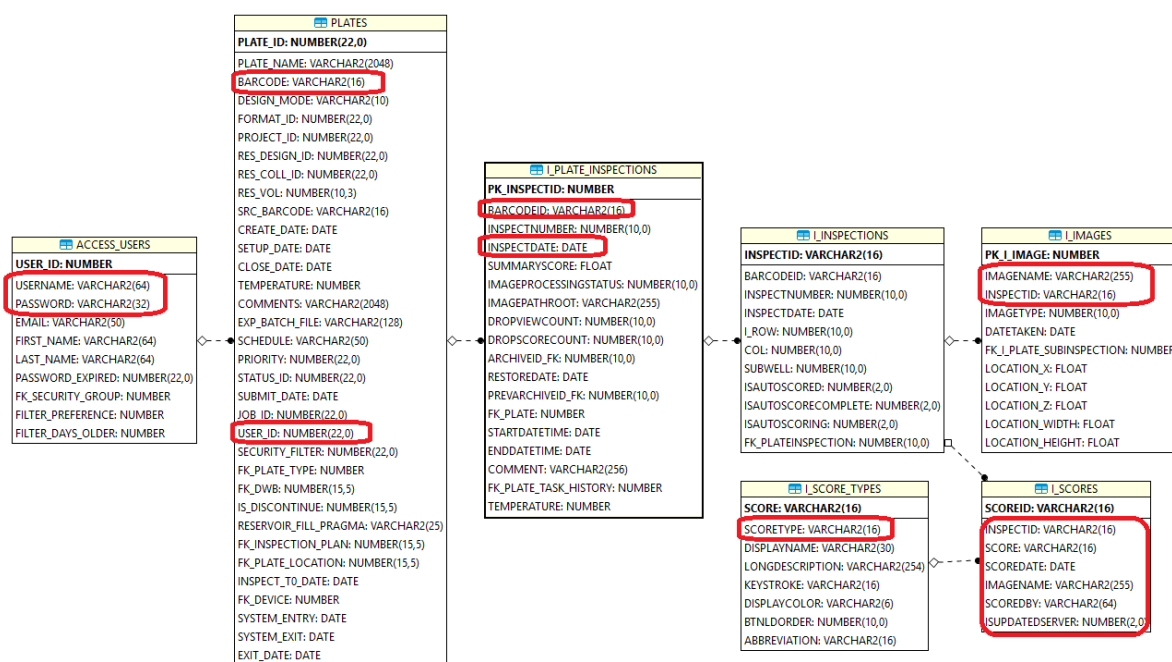


Figure S2 The relevant tables from C3's central database. This Oracle™ database is a slightly modified version of Rigaku's CM database. The tables which Cinder solo accesses contain more information that Cinder solo uses, the Cinder relevant rows are outlined in red in this image.