

JSON objects schemas

resources.json:

Array of either:

```
Repository: {  
  id: string;  
  name: string;  
  url: string;  
  created_date_timestamp: number;  
  type: "repository";  
  source: "github" / "bitbucket" / "gitlab";  
  last_push: string;  
  size: number;  
}
```

```
Image: {  
  id: string;  
  name: string;  
  url: string;  
  created_date_timestamp: number;  
  type: "image";  
  source: "dockerhub" / "ecr" / "jfrog";  
  number_of_layers: number;  
  architecture: "arm" / "amd";  
}
```

scans.json:

Array of:

```
Scan: {  
  scan_id: number;  
  resource_id: string;  
  resource_type: "repository" / "image";  
  highest_severity: "high" / "medium" / "low";  
  total_findings: number;  
  scan_date_timestamp: number;  
}
```

connections.json:

Array of:

```
Connection: {  
  repository_id: string;  
  image_id: string;  
}
```

Expected exported reconciliated records JSON schemas:

repository.json:

Array of:

```
Repository: {  
  id: string;  
  name: string;  
  url: string;  
  created_date_timestamp: number;  
  updated_date_timestamp: number;  
  scan_id: number;  
  highest_severity: string;  
  total_findings: number;  
  scan_date_timestamp: number;  
  connected_image_id: string;  
  source: "github" / "bitbucket" / "gitlab";  
  last_push: string;  
  size: number;  
}
```

image.json:

Array of:

```
Image: {  
  id: string;  
  name: string;  
  url: string;  
  created_date_timestamp: number;  
  updated_date_timestamp: number;  
  scan_id: number;  
  highest_severity: string;  
  total_findings: number;  
  scan_date_timestamp: number;  
  connected_repository_id: string;  
  source: "dockerhub" / "ecr" / "jfrog";  
  number_of_layers: number;  
  architecture: "arm" / "amd";  
}
```