

Migration of code from Imaris 6.0 to 6.1

1. Migration of Surfaces:

Components:

6.0: DataContainer, Surface

6.1: Surfaces

1.1. Get data (time, vertices, triangles, normals) from Imaris selection:

```
6.0: parent = imaris.mSurpassSelection; % parent must be of type IDataContainer
    for i = 0:parent.GetNumberOfChildren-1
        child = parent.GetChild(i); % child must be of type ISurface
        time = child.GetTimeIndex;
        vertices = child.GetVertices;
        triangles = child.GetTriangles;
        normals = child.GetNormals;
    end
```

```
6.1: parent = imaris.mSurpassSelection; % parent must be of type ISurfaces
    for i = 0:parent.GetNumberOfSurfaces-1
        time = parent.GetTimeIndex(i);
        vertices = parent.GetVertices(i);
        triangles = parent.GetTriangles(i);
        normals = parent.GetNormals(i);
    end
```

1.2. Set data (time, vertices, triangles, normals) to new objects:

```
6.0: parent = imaris.mFactory.CreateDataContainer;
    parent.mName = 'Surfaces';
    for i = 0:numel(time)-1
        child = imaris.mFactory.CreateSurface;
        child.SetSurface(vertices{i}, triangles{i}, normals{i}, time(i));
        child.mName = sprintf('Surface %d', i);
        parent.AddChild(child);
    end
    imaris.mSurpassScene.AddChild(parent);
```

```
6.1: parent = imaris.mFactory.CreateSurfaces;
    parent.mName = 'Surfaces';
    for i = 0:numel(time)-1
        parent.Add(vertices{i}, triangles{i}, normals{i}, time(i));
    end
    imaris.mSurpassScene.AddChild(parent);
```

2. Migration of Tracks

Components:

6.0: DataContainer, Tracks

6.1: Spots / Surfaces

2.1. Get positions and edges in one vector from Imaris selection:

```
6.0: parent = imaris.mSurpassSelection; % parent must be of type IDataContainer
positions = [];
edges = [];
for i = 0:parent.GetNumberOfChildren-1
    child = parent.GetChild(i); % child must be of type ITrack
    % map edges to the array of positions
    newEdges = child.GetEdges + size(positions, 1);
    edges = [edges, newEdges];
    % concatenate all positions and edges
    positions = [positions, child.GetSpots.GetPositionsXYZ];
end
```

6.1: Spots:

```
parent = imaris.mSurpassSelection; % parent must be of type ISpots
positions = parent.GetPositionsXYZ;
edges = parent.GetEdges;
```

Surfaces:

```
parent = imaris.mSurpassSelection; % parent must be of type ISurfaces
positions = zeros(parent.GetNumberOfSurfaces, 3);
for i = 0:parent.GetNumberOfSurfaces-1
    positions(i+1, :) = parent.GetCenterOfMass(i);
end
edges = parent.GetEdges;
```

2.2. Set positions, times and edges to new objects:

```
6.0: parent = imaris.mFactory.CreateDataContainer;
parent.mName = 'Tracks';
for i = 0:numel(times)-1
    spots = imaris.mFactory.CreateSpots;
    spots.Set(positions{i}, times{i}, radii{i});
    child = imaris.mFactory.CreateTrack;
    child.SetSpots(spots);
    child.SetEdges(edges{i});
    child.mName = sprintf('Track %d', i);
    parent.AddChild(child);
end
imaris.mSurpassScene.AddChild(parent);
```

6.1: Spots:

```
parent = imaris.mFactory.CreateSpots;
parent.mName = 'Tracks';
allPositions = [];
allTimes = [];
allRadii = [];
allEdges = [];
for i = 0:numel(times)-1
    % map edges to the array of positions
```

```
        newEdges = edges{i} + size(allPositions, 1);
        allEdges = [allEdges, newEdges];
        allPositions = [allPositions, positions{i}];
        allTimes = [allTimes, times{i}];
        allRadii = [allRadii, radii{i}];
    end
    parent.Set(allPositions, allTimes, allRadii);
    parent.SetTrackEdges(allEdges);
    imaris.mSurpassScene.AddChild(parent);
```