

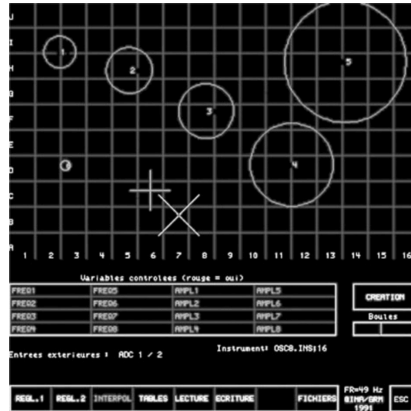
Faust Orbit UI

Yann Orlarey, EMERAUDE

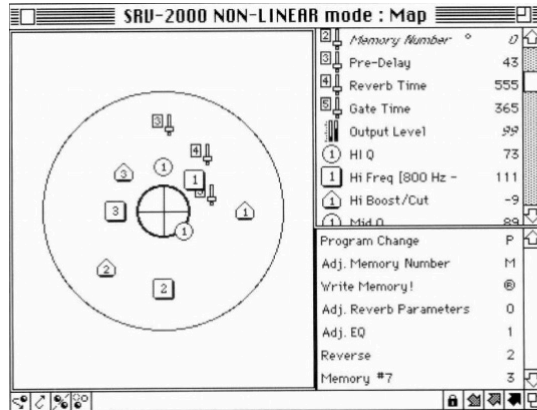
IFC 2026

2026-06-03

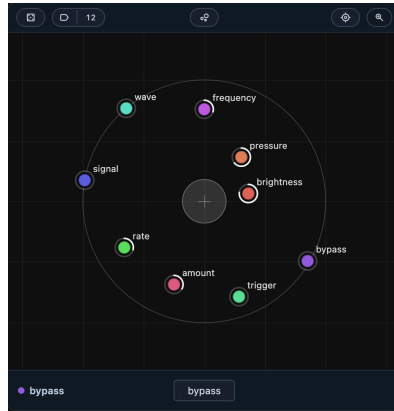
Origin : SYTER (GRM, 1984)



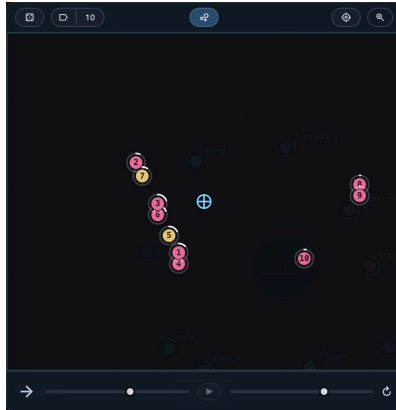
Legacy : INTER-ACTORS (TIMETECH/GAME, 1990)



Faust Orbit UI - Parameter Navigation Space



Faust Orbit UI - Preset Interpolation Space



Preset Interpolation Space - How is it built?

- PCA-based preset placement
- Shepard interpolation
- Trajectory loops

GitHub / npm Package

Where to find **faust-orbit-ui**?

- GitHub: <https://github.com/orlarey/faust-orbit-ui>
- npm: <https://www.npmjs.com/package/faust-orbit-ui>

Integration (1/3)

How to integrate **faust-orbit-ui**?

```
<div id="orbit-root" style="height: 480px"></div>
<link rel="stylesheet" href="./dist/faust-orbit-ui.css" />
<link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=
<script type="module">
  import { OrbitUI } from './dist/index.js';
```

Integration (2/3)

```
const ui = [  
  {  
    type: 'vgroup', label: 'synth',  
    items: [  
      { type: 'hslider', label: 'frequency', address: '/synth/freque  
        init: 440, min: 20, max: 2000, step: 1 },  
      { type: 'hslider', label: 'pressure', address: '/synth/pressur  
        init: 0.5, min: 0, max: 1, step: 0.01 },  
    ],  
  },  
];
```

Integration (3/3)

```
const orbitroot = document.getElementById('orbit-root');
const orbit = new OrbitUI(orbitroot, {
  uiDescriptor: ui,
  onParamChange: (path, value) => { // forward to your audio runtime
  onLibraryChange: (records) => idb.saveLibrary(orbit.uiHash, record
  onTrajectoryChange: (record) => idb.saveTrajectory(sessionId, reco
  onSelectionChange: (entries) => idb.saveSelection(sessionId, entri
  onLoopSettingsChange: (settings) => idb.saveLoopSettings(sessionId
});
</script>
```

Data Types

Config ::= Path → Value

Preset ::= (uiHash, configHash, name, time, **Config**)

Selection ::= position → (uiHash, configHash)

Loop ::= **Loop**(bpm, tp, level)

Event ::= **Commit**(time, **Config**)

Querying the Component

Method / property

`orbit.uiHash`

`orbit.getLibrary()`

`orbit.getSelection()`

`orbit.getTrajectory()`

`orbit.getLoopSettings()`

Returns

UI signature hash

Preset records

Ordered selection

Gesture history

BPM + portamento

Setting Component State

Method

`orbit.setParams(config)`

`orbit.setLibrary(records)`

`orbit.setSelection(entries)`

`orbit.setTrajectory(record)`

`orbit.setLoopSettings(settings)`

`orbit.setPromotionSuspended(true / false)`

Argument

Parameter values

Preset records

Ordered selection

Gesture history

BPM + portamento

Auto-promotion gate

Setters restore external state; they do not emit change callbacks.

Callbacks from the Component

Callback

`onParamChange(path, value)`

`onLibraryChange(records)`

`onSelectionChange(entries)`

`onTrajectoryChange(record)`

`onLoopSettingsChange(settings)`

`onCommit(configuration)`

`onInteractionStart() / onInteractionEnd()`

Payload / moment

Parameter changed

Preset library changed

Selection changed

Gesture history changed

BPM / portamento changed

Gesture committed

Gesture bracket