



JSON Forms 2.0

Form-based UIs for Web Applications

Maximilian Koegel
mkoegel@eclipsesource.com

Applications frequently require forms

Name*

Send email to Adrian

Due Date

Rating

3

Description

Confirm if you have passed the subject
Hereby ...

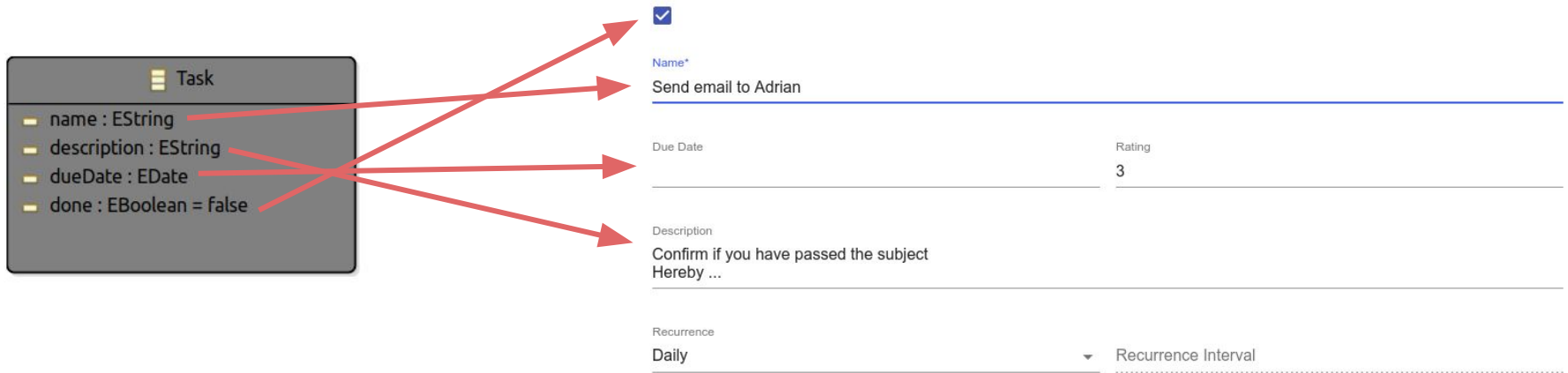
Recurrence

Daily



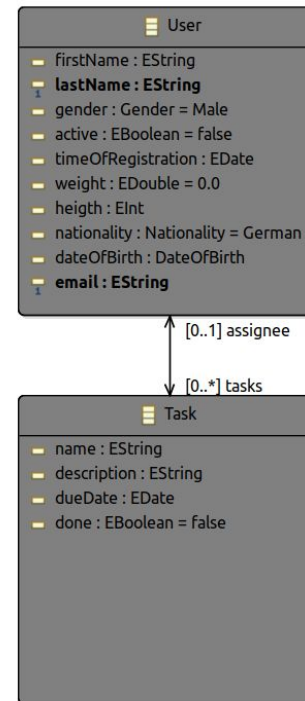
Recurrence Interval

Forms map data into a UI



Declarative Data Definition = JSON Schema

- “XML Schema for JSON”
- Format to describe JSON-based data
- Focused on data validation
- Can be created manually
- Can be generated from EMF Ecore

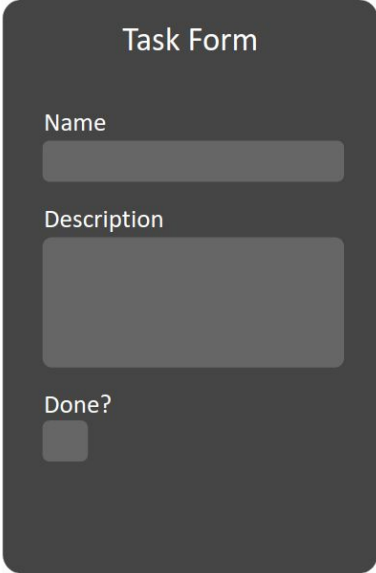


```
1 {
2   "type": "object",
3   "properties": {
4     "name": {
5       "type": "string",
6       "minLength": 1
7     },
8     "description": {
9       "type": "string"
10    },
11    "done": {
12      "type": "boolean"
13    },
14    "due_date": {
15      "type": "string",
16      "format": "date"
17    },
18    "rating": {
19      "type": "integer",
20      "maximum": 5
21    },
22    "recurrence": {
23      "type": "string",
24      "enum": [
25        "Never",
26        "Daily",
27        "Weekly",
28        "Monthly"
29      ]
30    }
31  }
32 }
```

→ **Used in JSON Forms to define the data**

Declarative UI Definition = UI Schema

- Declarative format for form-based UIs
- Describes the layout of a form
- Describes the data binding
- Does **not** describe the data
- References the data schema (JSON Schema)



The image shows a dark-themed UI mockup titled "Task Form". It contains three main sections: "Name" with a text input field, "Description" with a multi-line text area, and "Done?" with a checkbox. To the right of the mockup is a snippet of JSON code representing the UI schema. The code defines a "VerticalLayout" containing three "Control" elements. The first is a "HorizontalLayout" containing a "Control" for the "done" checkbox and another "Control" for the "rating" field. The second is a "Control" for the "name" field. The third is a "HorizontalLayout" containing a "Control" for the "description" field with a "multi" option set to true.

```
27 "type": "VerticalLayout",
28 "elements": [
29   {
30     "type": "Control",
31     "label": false,
32     "scope": "#/properties/done"
33   },
34   {
35     "type": "Control",
36     "scope": "#/properties/name"
37   },
38   {
39     "type": "HorizontalLayout",
40     "elements": [
41       {
42         "type": "Control",
43         "scope": "#/properties/due_date"
44       },
45       {
46         "type": "Control",
47         "scope": "#/properties/rating"
48       }
49     ]
50   },
51   {
52     "type": "Control",
53     "scope": "#/properties/description",
54     "options": {
55       "multi": true
56     }
57   }
58 ]
59 }
```

→ Used in JSON Forms to define the form-based UI

Combining JSON and UI Schema = Rendering

```

1 {
2   "type": "object",
3   "properties": {
4     "name": {
5       "type": "string",
6       "minLength": 1
7     },
8     "description": {
9       "type": "string"
10    },
11    "done": {
12      "type": "boolean"
13    },
14    "due_date": {
15      "type": "string",
16      "format": "date"
17    },
18    "rating": {
19      "type": "integer",
20      "maximum": 5
21    },
22    "recurrence": {
23      "type": "string",
24      "enum": [
25        "Never",
26        "Daily",
27        "Weekly",
28        "Monthly"
29      ]
30    },

```



```

1 s{
2   "type": "VerticalLayout",
3   "elements": [
4     {
5       "type": "Control",
6       "label": false,
7       "scope": "#/properties/done"
8     },
9     {
10      "type": "Control",
11      "scope": "#/properties/name"
12    },
13    {
14      "type": "HorizontalLayout",
15      "elements": [
16        {
17          "type": "Control",
18          "scope": "#/properties/due_date"
19        },
20        {
21          "type": "Control",
22          "scope": "#/properties/rating"
23        }
24      ]
25    },
26    {
27      "type": "Control",
28      "scope": "#/properties/description",
29      "options": {
30        "multi": true
31      }
32    },
33  ]

```



Name*
Prepare slides for demo

Due Date Rating
2018-03-27 8

should be <= 5

Description
This is some awesome multi-line text to show that it may or may not make sense to display it this way

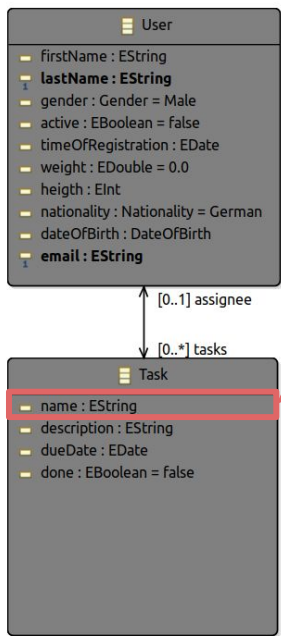
Recurrence
Never ▼ Recurrence Interval

**JSON Schema
(Data Schema)**

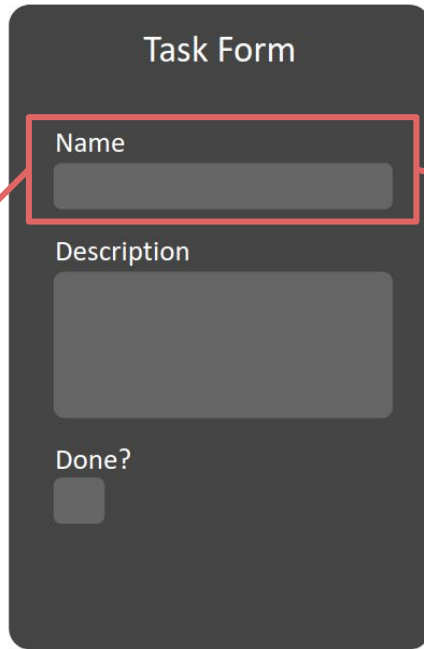
UI Schema

User Interface

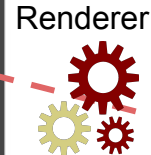
Rendering Controls



**JSON Schema
(Data Schema)**



UI Schema



Name*

Prepare slides for demo

Due Date

2018-03-27

Rating

8

should be <= 5

Description

This is some awesome multi-line text to show that it may or may not make sense to display it this way

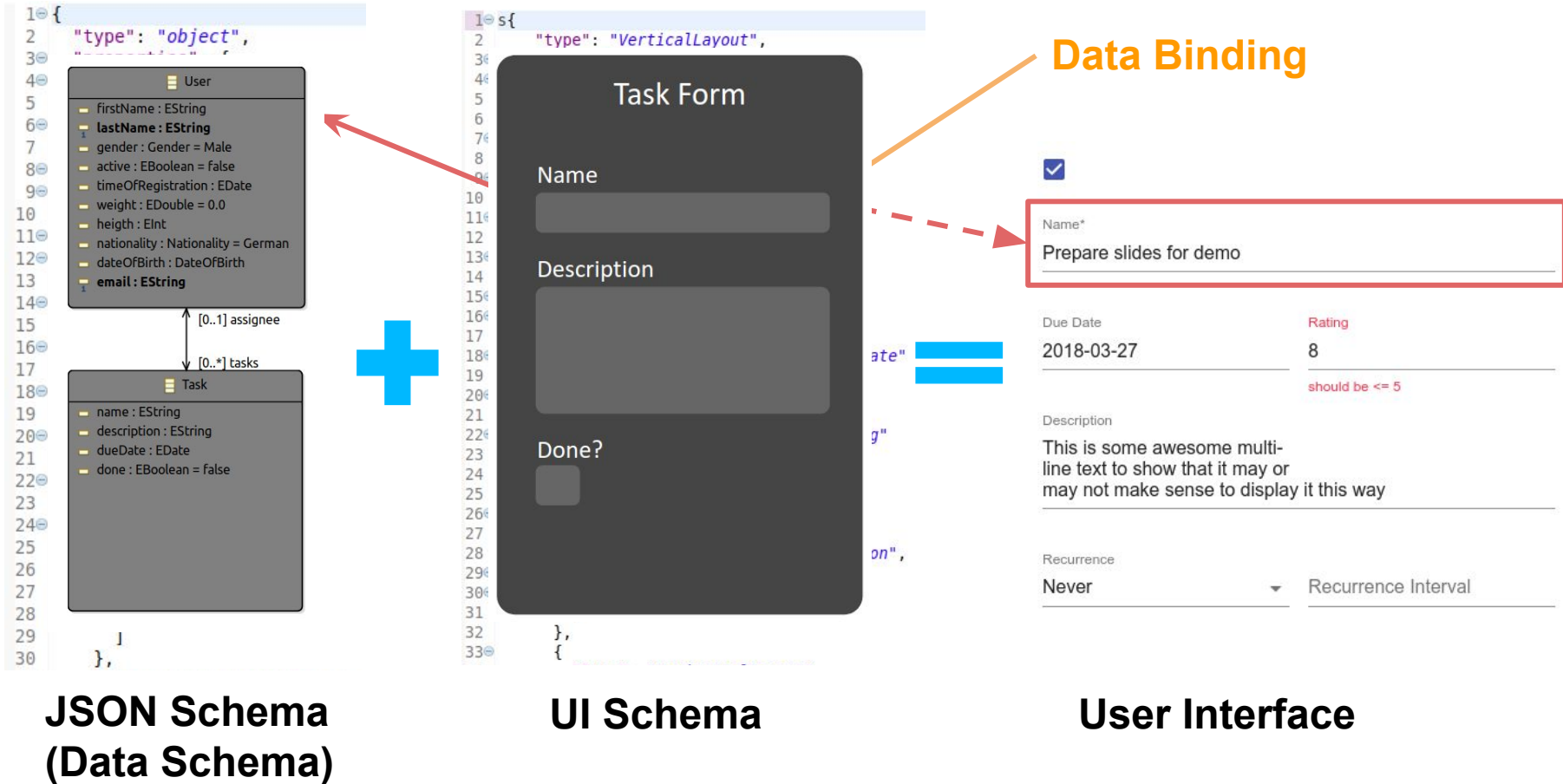
Recurrence

Never

Recurrence Interval

User Interface

Rendering Controls



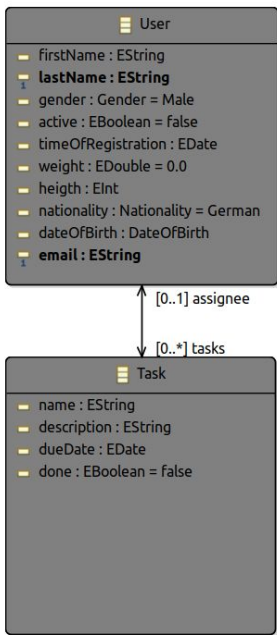
Data Binding

**JSON Schema
(Data Schema)**

UI Schema

User Interface

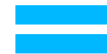
Rendering Layouts



**JSON Schema
(Data Schema)**

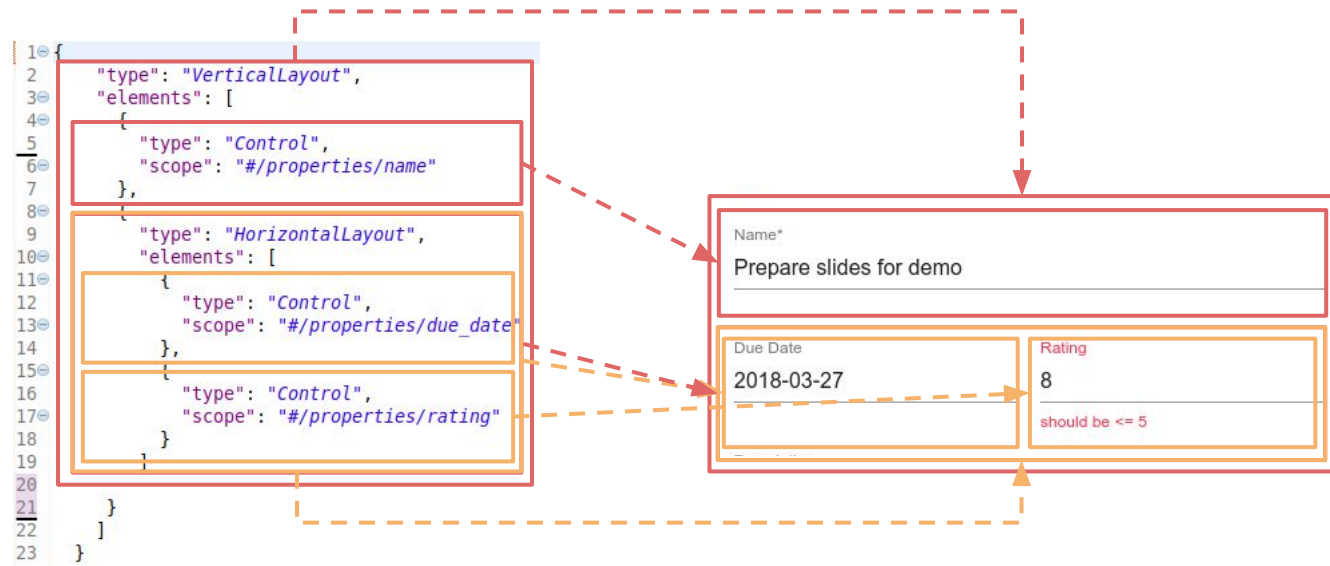


UI Schema



User Interface

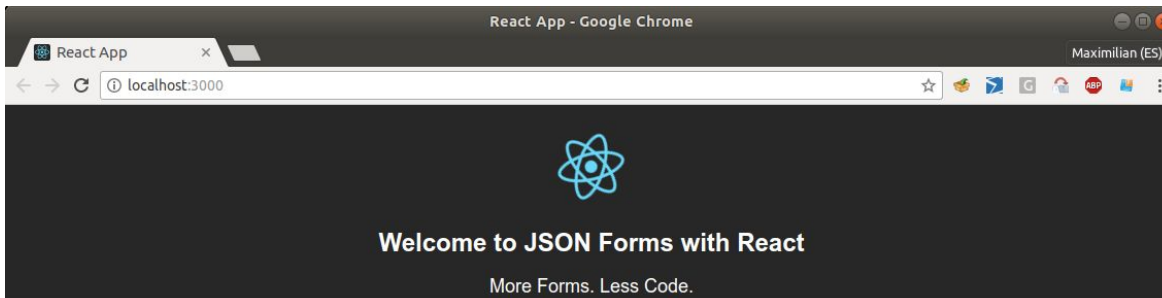
Rendering Layouts



UI Schema

User Interface

Demo!



Bound data

```
{
  "name": "Send email to Adrian",
  "description": "Confirm if you have passed the subject\nHereby ...",
  "done": true,
  "recurrence": "Daily",
  "rating": 3
}
```



Name*

Send email to Adrian

Due Date

Rating

3

Description

Confirm if you have passed the subject
Hereby ...

Recurrence

Daily



Recurrence Interval

Demo: Validation



Name*

should NOT be shorter than 1 characters

Due Date

2018-03-27

Rating

8

should be <= 5

Description

This is some awesome multi-line text to show that it may or may not make sense to display it this way

Recurrence

Never



Recurrence Interval

```

1 {
2   "type": "object",
3   "properties": {
4     "name": {
5       "type": "string",
6       "minLength": 1
7     },
8     "description": {
9       "type": "string"
10    },
11    "done": {
12     "type": "boolean"
13    },
14    "due_date": {
15     "type": "string",
16     "format": "date"
17    },
18    "rating": {
19     "type": "integer",
20     "maximum": 5
21    },
22    "recurrence": {
23     "type": "string",
24     "enum": [
25       "Never",
26       "Daily",
27       "Weekly",
28       "Monthly"
29     ]
30  },

```

Demo: Visibility Rules

Recurrence
Daily



Recurrence Interval
2

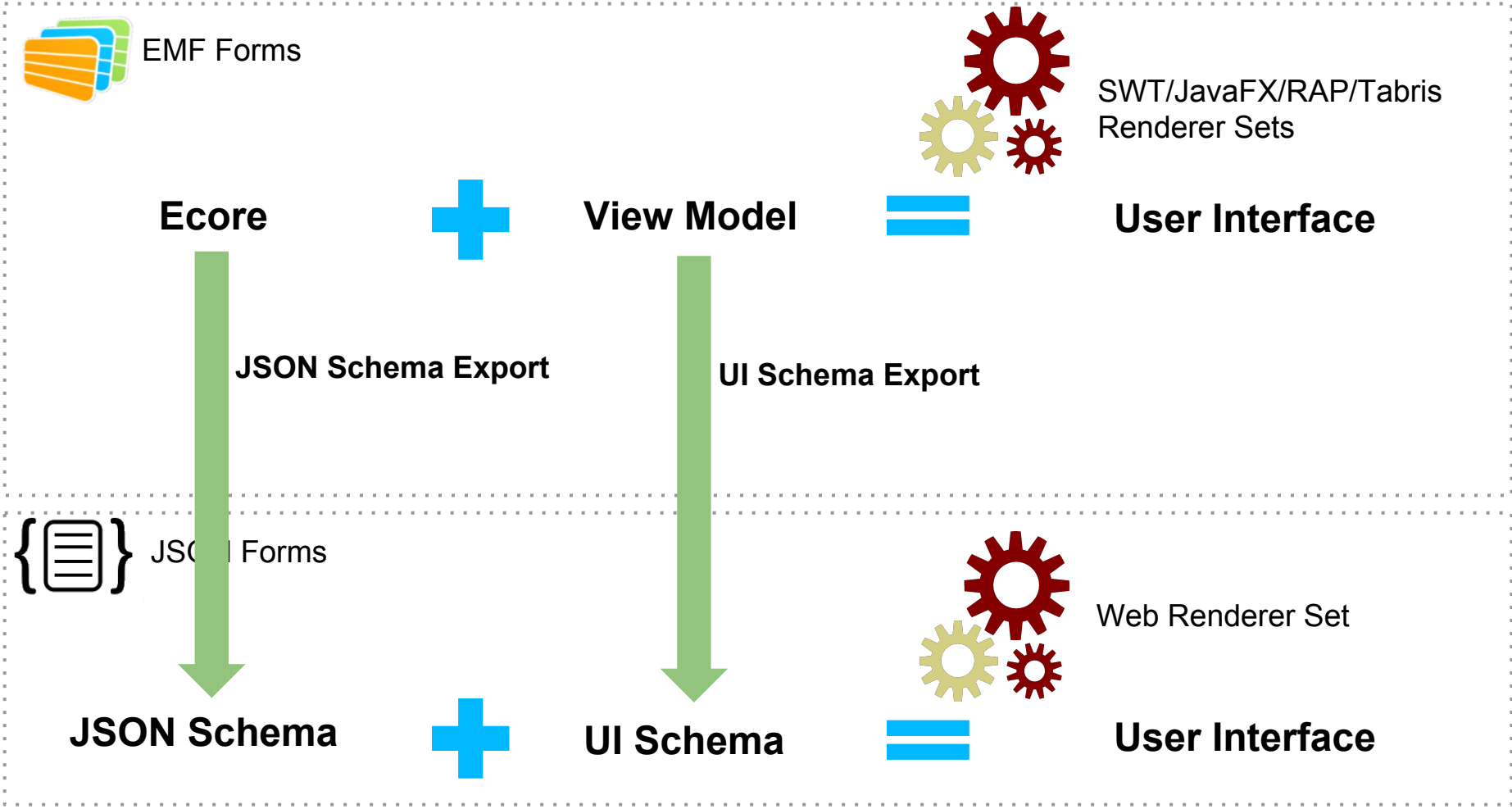
Demo: Custom Renderers

Rating

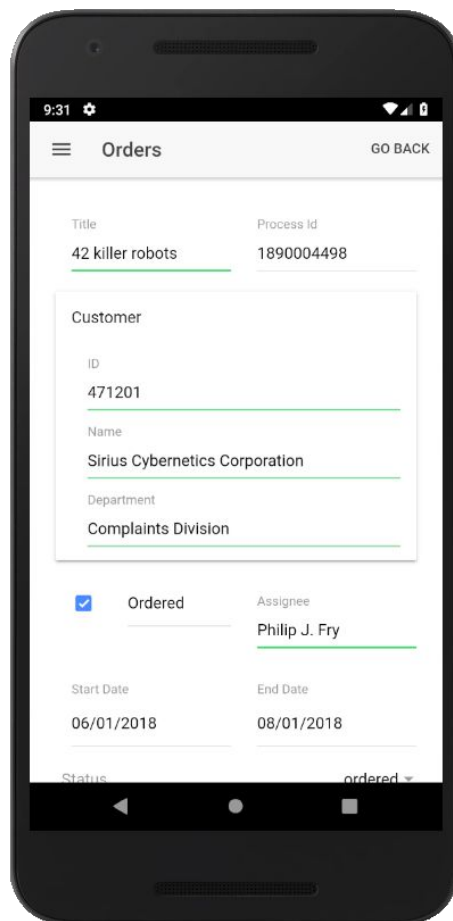
3



Demo: EMF Forms Integration



Demo: Mobile Renderer with Ionic



What's new with 2.0? (1/2)

- Improved Architecture
 - Redux-based architecture
 - Renderers are simpler (no update handling)
- Renderer Sets
 - Material React and Vanilla React
 - Ionic (WIP)
 - Angular Material (WIP)
 - Easy to create new one: 20-50 days

What's new with 2.0? (2/2)

- Improved Schema support:
 - Any of & All of
 - Reference resolving
- Support for Trees:
 - TreeWithDetail
 - Tree-Editor
- Eclipse Theia Integration: Theia-Tree-Editor

More Information

JSON Forms

Web: <http://jsonforms.io>

Twitter: #jsonforms @JSONForms



EMF Forms

Web: <http://emfforms.org>

Twitter: #emfforms @EMFForms



EclipseCon Europe

Building Web-based Modeling Tools based on Theia - 11:30 AM - Theater Stage

The Busy RCP Developer's Guide to Eclipse Theia - 1:15 PM - Theater Stage



Evaluate **this** Session

Sign in and vote at eclipsecon.org

WITH

~~-1~~

~~0~~

+1