In [21]:
```
import pandas as pd
df = pd.read_csv(r'C:\Users\giral\OneDrive\Documentos\Master_Ciencias_de_Datos\Visualizacion\PEC2\imre12327_Supplemental_Material\gf_imr.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	stock	demo	sex	year0	interval	orig	dest	orig_code	dest_code	flow
0	un12	wpp2010	b	1990	10	ABW	ABW	533	533	0
1	un12	wpp2010	b	1990	10	ABW	AFG	533	4	0
2	un12	wpp2010	b	1990	10	ABW	AGO	533	24	0
3	un12	wpp2010	b	1990	10	ABW	ALB	533	8	0
4	un12	wpp2010	b	1990	10	ABW	ANT	533	530	0

In [13]:
```
import numpy as np
```

In [14]:
```
# Eliminamos las variables que no nos interesan
df = df.drop('stock', 1)
df = df.drop('demo', 1)
df = df.drop('sex', 1)
df = df.drop('year0', 1)
df = df.drop('interval', 1)
df = df.drop('orig_code', 1)
df = df.drop('dest_code', 1)
```

In [15]:
```
df.head()
```

Out[15]:

	orig	dest	flow
0	ABW	ABW	0
1	ABW	AFG	0
2	ABW	AGO	0
3	ABW	ALB	0
4	ABW	ANT	0

In [16]:
```
df1 = df[df["flow"]!=0] # Filtramos los 0, porque nos interesa que exista flujo
df1 = df1.head(210) # Filtramos 210 registros para visualizarlos
df1.head()
```

Out[16]:

	orig	dest	flow
9	ABW	AUT	1
29	ABW	CAN	2
30	ABW	CHE	1
45	ABW	DEU	56
47	ABW	DNK	1

In [17]:
```
import holoviews as hv
from holoviews.core import Store
import pandas as pd

hv.ipython.notebook_extension('bokeh')

Store.set_current_backend('bokeh')
renderer = Store.renderers['bokeh']

#df_final = pd.DataFrame({
#    'Sub_Market': ['Central texas', 'Southern California', 'Florida'],
#    'Sport League': ['MLS', 'NBA', 'MLS'],
#    'Revenue': [1.4981211 * 10**5, 2.921212* 10**6, 1.2121112*10**6]
#})

graph = hv.Sankey(
    df1, #df_final-->df1
    kdims=['orig', 'dest'], # Sub_Market --> orig /// Sport League --> dest
    vdims=[hv.Dimension("flow", value_format=str)], # Revenue --> flow
)
```

In [18]:
```
renderer = Store.renderers['bokeh']
plot = renderer.get_plot(graph)
```

In [19]:
```
offset = -200
num_nodes = len(plot.handles['text_1_source'].data['x'])
plot.handles['text_1_source'].data['x_offset'] = [0]* num_nodes
num_left_nodes = 3
left_nodes_selection = slice(0, num_left_nodes)
plot.handles['text_1_source'].data['x_offset'][left_nodes_selection] = [offset]* num_left_nodes
plot.handles['text_1_glyph'].x_offset = {'field': 'x_offset' }
plot.handles['plot'].x_range.start += (2*offset)
```

In [20]:
```
hv.ipython.notebook_extension('bokeh')
data, metadata = hv.ipython.display_hooks.render(plot, fmt='svg')
hv.ipython.display(hv.ipython.HTML(data["text/html"]))
```


In [ ]:
```
```