

JavaScript Front-End Web App Tutorial Part 5: Managing Bidirectional Associations

**Learn how to manage bidirectional
associations between object types,
such as between books and their
authors, and authors and their books**

Gerd Wagner <G.Wagner@b-tu.de>

JavaScript Front-End Web App Tutorial Part 5: Managing Bidirectional Associations: Learn how to manage bidirectional associations between object types, such as between books and their authors, and authors and their books

by Gerd Wagner

Warning: This tutorial manuscript may still contain errors and may still be incomplete in certain respects. Please report any issue to Gerd Wagner at G.Wagner@b-tu.de.

This tutorial is also available in the following formats: PDF [[bidirectional-association-tutorial.pdf](#)]. You may run the example app [[BidirectionalAssociationApp/index.html](#)] from our server, or download it as a ZIP archive file [[BidirectionalAssociationApp.zip](#)]. See also our Web Engineering project page [<http://web-engineering.info/>].

Publication date 2015-11-12

Copyright © 2014-2015 Gerd Wagner

This tutorial article, along with any associated source code, is licensed under The Code Project Open License (CPOL) [<http://www.codeproject.com/info/cpol10.aspx>], implying that the associated code is provided "as-is", can be modified to create derivative works, can be redistributed, and can be used in commercial applications, but the article must not be distributed or republished without the author's consent.

Table of Contents

Foreword	vi
1. Bidirectional Associations	1
1. Inverse Reference Properties	1
2. Making an Association-Free Information Design Model	3
2.1. The basic procedure	3
2.2. How to eliminate unidirectional associations	3
2.3. How to eliminate bidirectional associations	3
2.4. The resulting association-free design model	4
2. Implementing Bidirectional Associations with Plain JavaScript	6
1. Make a JavaScript Data Model	6
2. Write the Model Code	7
2.1. New issues	7
2.2. Summary	8
2.3. Encode each class of the JavaScript data model as a constructor function	9
2.4. Encode the property checks	9
2.5. Encode the setter operations	9
2.6. Encode the add and remove operations	10
2.7. Take care of deletion dependencies	11
3. Exploiting Derived Inverse Reference Properties in the User Interface	11
3.1. Show information about published books in the <i>List Publishers</i> use case	11
4. Practice Project	12

List of Figures

1.1. The Publisher-Book-Author information design model with two bidirectional associations	1
1.2. Turn a bidirectional one-to-one association into a pair of mutually inverse single-valued reference properties	4
1.3. Turn a bidirectional many-to-many association into a master-slave pair of mutually inverse multi-valued reference properties	4
1.4. The association-free design model	5
2.1. The JavaScript data model without the class <code>Book</code>	7
2.2. Two bidirectional associations between <code>Movie</code> and <code>Person</code>	12

List of Tables

2.1. Sample data for Publisher	6
2.2. Sample data for Book	6
2.3. Sample data for Author	7
2.4. Movies	12
2.5. People	12

[Click here to download full PDF material](#)