Approaches to Two-Way Databinding

*Author:* [*rob@bluespire.com*](mailto:rob@bluespire.com)

This document attempts to catalog various strategies for accomplishing two-way databinding in modern and near-future web applications. By doing so, we can better select a technique for AngularJS 2.0.

# Approaches

## Dirty Checking (Multi-Digest)

## Dirty Checking (Limited Scope)

## Explicit Observable Model

## Implicit Observable Model

## Hybrid Implicit Observable and Explicit Observable

## Hybrid Dirty Checking & Explicit Observable

## Hybrid Implicit Observable and Dirty Checking

## Hybrid Implicit Observable, Explicit Observable and Dirty Checking

***Note:*** *Techniques involving both Observables and Dirty Checking can be looked at from multiple technical viewpoints. For example, observables can be layered on top of dirty checking. In this world, whenever an observable wishes to notify its subscribers, it inserts itself itself into the dirty record list. The next time a digest occurs, the record will be processed as normal, then removed from the record list. However, if we look at the world from the perspective of observables first, we arrive at a different solution. In this world, observables notify as expected (preferably batching on the micro task queue) but dirty checking runs separately. However, the user API for dirty checked properties looks identical to an observable.*

# Summary