Document Scope

This document outlines a change required to an existing prebid setup in order for a publisher to host an additional pbjs build in parallel. This change should only be made where the dual setup is in use.

Suggested Method

The existing pbjs build will require an edit to the standard code that's being used.

The pbjs instance of bidsBackHandler will typically call a function - initAdServer - which handles set the targeting keys and tells DFP to fetch ads. EG:

```
pbjs.que.push(function() {
    pbjs.addAdUnits(adUnits);
    pbjs.requestBids({
        bidsBackHandler: initAdserver,
        timeout: PREBID_TIMEOUT
    });
});

function initAdserver() {
    if (pbjs.initAdserverSet) return;
    pbjs.initAdserverSet = true;
    googletag.cmd.push(function() {
        pbjs.que.push(function() {
            pbjs.setTargetingForGPTAsync();
            googletag.pubads().refresh();
        });
    });
}
// in case PBJS doesn't load
setTimeout(function() {
    initAdserver();
}, FAILSAFE_TIMEOUT);
```

The initAdserver function will need to be amended so that it no longer calls googletag.pubads().refresh but calls our custom function (pbjsSync) so that both prebid instances targeting can be returned before the page attempts to call DFP/GPT - for example:

```
pbjs.que.push(function() {
    pbjs.addAdUnits(adUnits);
    pbjs.requestBids({
        bidsBackHandler: initAdserver,
```

```
            timeout: PREBID_TIMEOUT
        });
    });

    function initAdserver() {
        if (pbjs.initAdserverSet) return;
        pbjs.initAdserverSet = true;
        googletag.cmd.push(function() {
            pbjs.que.push(function() {
                if(!window.pbjsSync) {
                console.error('pbjs ERROR: no pbjsSync object found');
                }
                pbjs.setTargetingForGPTAsync();
                pbjsSync.registerTargetingSet('pbjs');
            });
        });
    }
    // in case PBJS doesn't load
    setTimeout(function() {
        initAdserver();
    }, FAILSAFE_TIMEOUT);
```