

Contents

- [1 Introduction](#)
- [2 A leg vs B leg](#)
- [3 One legged calls](#)
- [4 Bridged calls](#)
- [5 Propogating variables from the A leg to the B leg](#)

Introduction

Call legs describe active pieces of a call. The connections from a SIP provider to FreeSWITCH or from FreeSWITCH to a OpenZAP channel are examples of a call leg. From FreeSWITCH's point of view, there are *legs* going to each party of the call. Looking at it that way, call legs could be over-simplified as representing each half of the call.

In FreeSWITCH, each call leg has a UUID that can be used programatically to manipulate the call, or reported on in call detail records.

A leg vs B leg

The terminology of **A Leg** and **B Leg** can sometimes be confusing. In the strictest technical sense, the A leg represents *ingress* to the switch while the B leg represents *egress* from the switch. In most cases this means that the originator of the call is the A leg and the recipient of the call is the B leg.

One legged calls

Calls that are in an IVR, checking voicemail, etc - ones that don't have an egress leg, are sometime called one legged calls. In this scenario, the switch itself acts as the B leg.

Bridged calls

Once an A leg is connected to a B leg, the channels are **bridged**. Usually this means that if one leg hangs up the other leg continues with the dialplan (unless it doesn't have any more actions, or a dialplan at all). Bridges can be broken in other ways too (like intercept, or transferring one leg somewhere else, or bridging a leg to something else). Variables that affect what happens to the other leg when the bridge ends include hangup after bridge, park after bridge and transfer after bridge.

Propogating variables from the A leg to the B leg

Sometimes you want to propogate variables from the A leg to the B leg or set new variables there. From the dialplan this can be done via the export command.