SIP forking refers to the process of “forking” a single SIP call to multiple SIP endpoints. This is a very powerful feature of SIP. A single call can ring many endpoints at the same time.

With SIP forking, you can have your desk phone ring at the same time as your softphone or a SIP phone on your mobile, allowing you to take the call from either device easily. No forwarding rules would be necessary as both devices would ring.

In the same manner, SIP forking can be used in an office and allow the secretary to answer calls to the extension of his/her boss when he is away or unable to take the call.

Note that a forking SIP proxy cannot be stateless because it needs to perform a filtering operation, returning one response out of the many it receives.

We have two types of forking:

- Parallel Forking
- Sequential Forking

**Parallel Forking**

In this scenario, the proxy server will fork the INVITE to, say, two devices UA2, UA3 at a time. Both the devices will generate 180 Ringing and whoever receives the call will generate a 200 OK. The response **suppose UA2** that reaches the Originator first will establish a session with UA2. For the other response, a CANCEL will be triggered.

If the originator receives both the responses simultaneously, then based on q-value, it will forward the response.

**Sequential Forking**
In this scenario, the proxy server will fork the INVITE to one device UA2. If UA2 is unavailable or busy at that time, then the proxy will fork it to another device UA3.

**Branch - ID & TagID**

Branch IDs allow proxies to match responses to forked requests. Without them, a proxy wouldn’t be able to tell which branch a response corresponds to.

Tags are used by the UAC to distinguish multiple final responses from different UAS. A UAS has no reliable way of determining if the request has been forked or not. To be safe, it needs to add a tag. Proxies only insert tags into the final responses they generate themselves; they never insert tags into requests or responses they forward.

Since a request can be forked several times on its way to UAS, a single “tag” added to the request by one of the proxies is not sufficient for the next forking proxy along the chain to match responses on its own branches; every proxy that forked the request would need to add its own unique IDs to the branches it created.

**Call leg & Call ID**

A call leg refers to one to one signalling relationship between two user agents. The call ID is an identifier carried in SIP message that refers to the call. A call is a collection of call legs.

A UAC starts by sending an INVITE. Due to forking, it may receive multiple 200 OK from different UAs. Each corresponds to a different call leg within the same call.

A call is thus a group of call legs. A call leg refers to end-to-end connection between UAs.

The CSeq spaces in the two directions of a call leg are independent. Within a single direction, the sequence number is incremented for each transaction.
Voicemail

Voicemail is a messaging service commonly associated with telephony applications. It can be implemented as a service in a network as provided by mobile phone providers, in a separate device such as a home answering machine, or incorporated in a telephony device such as an enterprise PBX or key system.

Voicemail involves call forwarding no answer/busy/unavailable to a storage device which plays a customizable greeting. The user is then alerted by some means that a message is waiting, and can then retrieve the message by dialling into the system.

In SIP terms, the call forwarding is straightforward, with either a proxy forwarding or endpoint redirection 3xxresponse used to send the call to the voicemail server. However, some kind of SIP extension is needed to indicate to the voicemail system which mailbox to use—that is, which greeting to play and where to store the recorded message. We have two ways to do this:

- Use a SIP header field extension
- Use the Request-URI to signal this information

For a voicemail system at sip:voicemail.example.com, which is being used to provide voicemail for sip:alice@example.com, the Request-URI of the INVITE when it is forwarded to the voicemail server could look like:

```
sip:voicemail.example.com; target = sip:alice@example.com; cause = 486
```

In this way, the Request-URI carries the mailbox identifier as well as the reason the call is being forwarded to the voicemail.
SIP Voicemail Callflow

Voicemail server answers, plays UA2 greeting, records message, then hangs up with UA1

UA2 is not registers and subscribes to get MVI

Message Waiting Indication is rendered to user