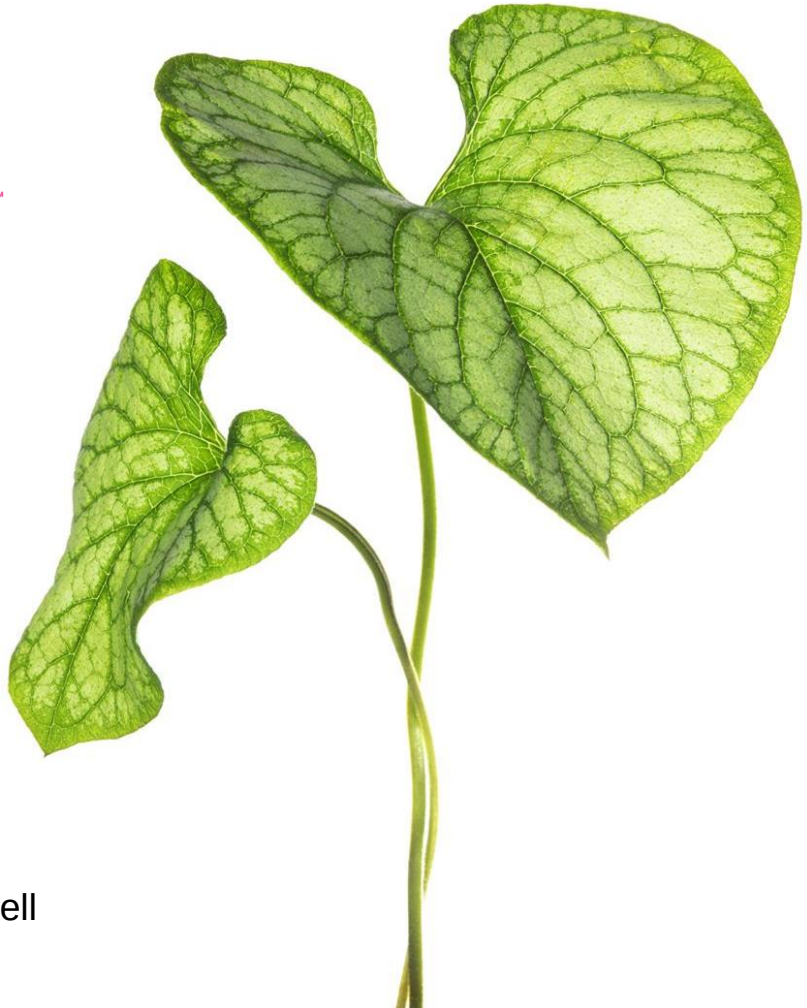
Fragmented carriers in the DL

A proposal for Rel-19 study item

RAN4 #110

TELUS, Bell Mobility, T-Mobile USA, Telstra Limited,
US Cellular Corporation, AT&T, Nokia, Nokia Shanghai Bell



Background

- The maximum number of component carriers (CCs) in a Carrier Aggregation (CA) combo is limited by the number and capabilities of the analog Rx chains in the UEs
- As a result, it is not expected that the number of CCs that chipsets support will go up significantly in the coming years (6 or 7 is likely the limit in the FR1 range)
- Today, fragmented carriers in the same band consume multiple Rx chains, thus limiting the CA capabilities
- We propose studying the use of fewer Rx chains than carrier fragments in a band which could result in freeing up Rx chains for additional component carriers
- This restriction exists only in the RF domain as some chipset vendors have already demonstrated that fragmented blocks require the same number of baseband processing units as single non-fragmented blocks (e.g., 7A-7A requires single downlink processing unit, same as 7A)

General

Example of fragmented blocks in FR1 mid-bands in Canada

PCS (n25) example of fragmented spectrum in Toronto*

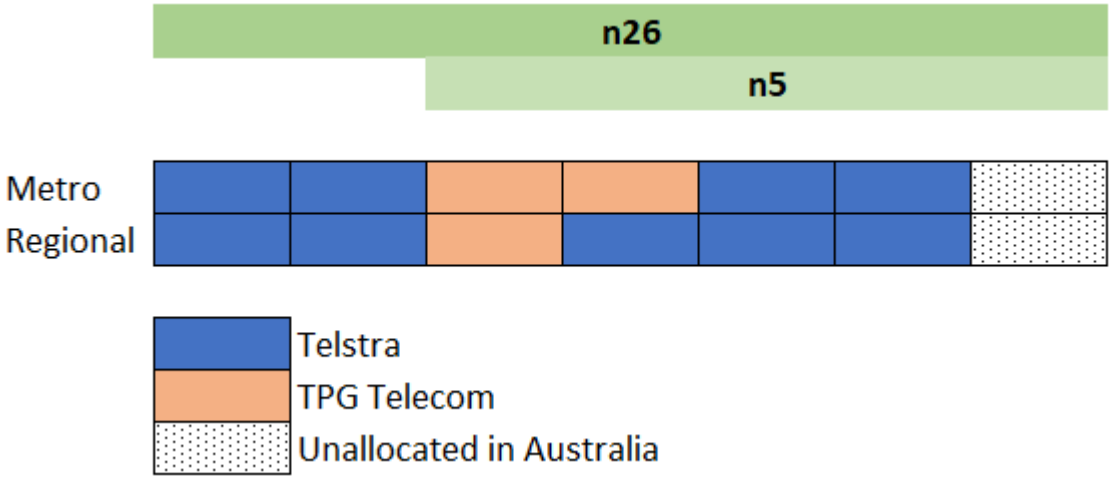
BRS (n7) example of fragmented spectrum in Toronto*

AWS1/3/4 (n66) example of fragmented spectrum in Toronto*

*Same colour indicates spectrum access for the same operator; each block is 5 MHz wide

General

Example of fragmented blocks in FR1 low-band in Australia



*Same colour indicates spectrum access for the same operator; each block is 5 MHz wide

General

Study Item Proposals

1. The feasibility of dynamically determining the number of Rx chains needed per band of a CA combo
2. Backward compatibility concerns
3. The need for additional signalling
4. Limit the scope to individual CC DL bandwidth that are ≤ 100 MHz (include FDD and TDD)

General

Proposed solution example

- In the example below, currently 6 Rx chains are needed, thus precluding additional CCs to be added (assuming a 6 Rx limit)
- Our proposal is to implement this dynamically using 3 to 6 Rx chains
- This would enable 10+ CCs combo implementation for much higher speeds

Current count:	2 Rx / 2 CCs	2 Rx / 2 CCs	2 Rx / 2 CCs	**Total:** 6 Rx chains / 6 CCs
Proposed count:	N_1 Rx / 2 CCs $N_1 = 1 \text{ or } 2$	N_2 Rx / 2 CCs $N_2 = 1 \text{ or } 2$	N_3 Rx / 2 CCs $N_3 = 1 \text{ or } 2$	**Total:** $(N_1 + N_2 + N_3)$ Rx chains / 6 CCs

General