Branch-Based & Flag-Based

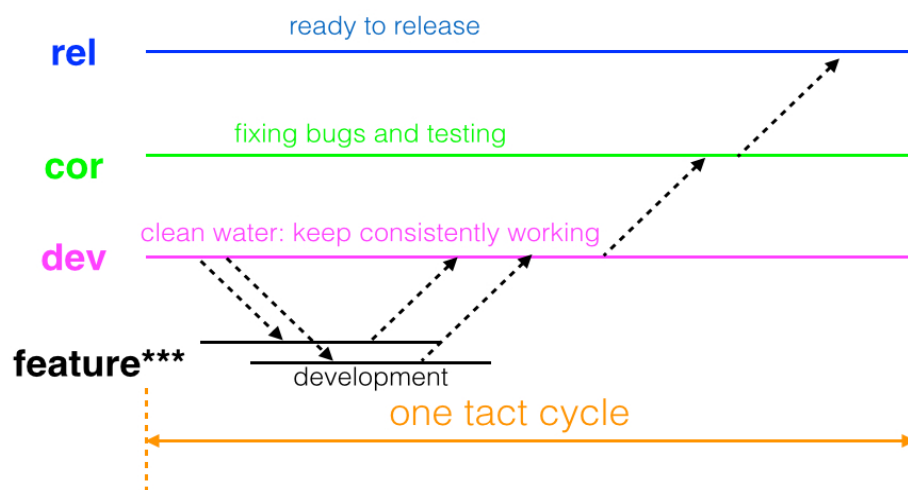|	Group	Goal	Feature flag-based		Corresponding branch-based	
			KPI	Rationale	KPI	Rationale
1	One code line (From Gerrit) The deviation of code: how quickly to get the latest code in/out? To reduce the unintegrated code to master.	To minimize the time reaching to release	How soon does merge happen?	Average TAT for all/each subtopic branch (Turn Around Time)	Which branch is the bottleneck?	The ~~maximum~~/average single life cycle among all feature branches (from "Start of Dev" until "End of Dev")
		To minimize the integration effort	How often are the successful merges?	Success rates (# of commits merged into master branch/# non-trivial patchset)	How difficult to merge a commit?	Size of merge commit (Indication of effort) (Bram thinks not same metric as for feature-based but Josh thinks they both refer to integration effort)
		To shorten the integration time	How long it takes to merge subtopic branch into master?	Average build time on master branch (total build time/# of non-trivial patch-sets)	How long it takes to merge feature branch into dev?	The average build time on dev branch (total build time/# feature branches)
2	High velocity in Development (From Git) High, consistent velocity in commits	To decrease the size of commits	How large are commits?	Average LOC of commits	X (X means the same as feature-based)	X
		To shorten the integration of new commits	How long does it take to integrate a commit to subtopic branch?	Average build time per commit	X	X
		To increase the number	How many commits got merged	Success rate per subtopic/feature branch	X	X

		of merged commits per day	into each branch per day?			
3	Frequent delivery (From JIRA) (NA) (can only be accessed from Orca end)	To increase the productive commits	How many commits got merged into production line?	# of commits merged to stabi/cor branch	X	X
		To decrease the amount of useless cherry-picks	How many cherry-picks are really merged?	Regression rate of cherry-pick fixes	X	X
		To decrease the number of defects	How many defects are fixed?	Incoming and outgoing defect rate	X	X

Flag-based Development Process

stabi ——————— ready to release ————————————

master ——— build + test + fix bugs ————————————

subtopic*** ———— development ————

one wave
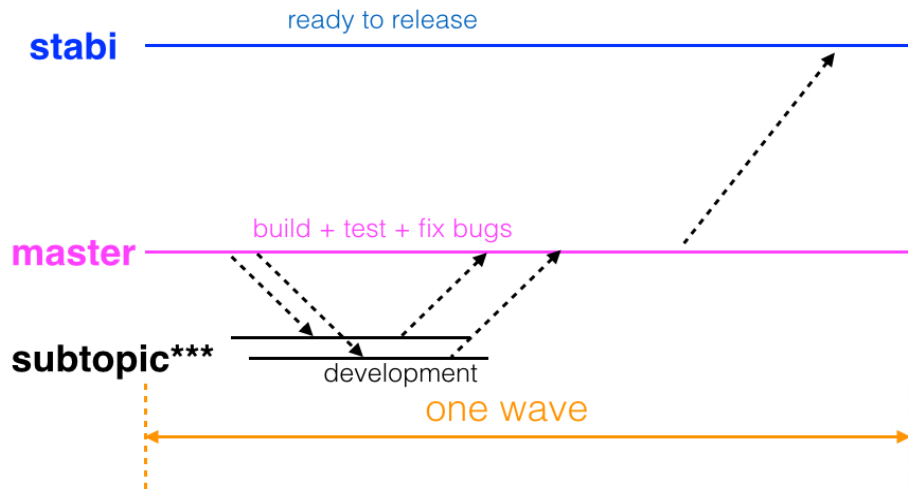
Each feature branch works separately. Check in to the master branch at anytime. If not complete, set the flag "off". Until all subtopic branches set the flag "on". One turnaround is done.

Branch-based Development Process

rel ——————— ready to release ————————————

cor ——— fixing bugs and testing ————————————

dev — clean water: keep consistently working ——————

feature*** ———— development ————

one tact cycle

Each feature branch works on its own task separately. Check in to the dev branch until the whole feature is completed.