

Background

When working on JIRA <https://issues.apache.org/jira/browse/FLINK-33977> : during adaptive rescaling, what strategy should be used to select candidate slots to ensure efficient/expected resource utilization? We have received some lively discussions and valuable feedback:

1. When deploying jobs with an Application cluster, there seems to be a preference for using the fewest TaskManagers.
2. When deploying with a Session cluster, a slot spread-out strategy is favored to achieve more balanced resource usage.
3. Some have also suggested introducing configurable parameters to specify strategy priorities, giving users more flexibility when there are conflicts between different strategies. like <https://issues.apache.org/jira/browse/FLINK-36426>
4. more discussion details or valuable ideas could be viewed here <https://github.com/apache/flink/pull/25218#issuecomment-2401913141>

Therefore, we need a discussion to clarify these matters. It's worth mentioning that this discussion should also include **taskmanager.load-balance.mode=SLOTS** <https://issues.apache.org/jira/browse/FLINK-33390> for two main reasons: first, the current discussions have already highlighted this issue, and second, this aspect will soon be introduced with the upcoming JIRA changes.

Based on the existing discussions and preferences, I have summarized some conflicting items and proposed simple alternatives if no new configuration items are introduced, hoping to provide a little value in advancing the current work.

Slot assignment priority conflicts

when introducing minimal-taskmanagers and **taskmanager.load-balance.mode=SLOTS** <https://issues.apache.org/jira/browse/FLINK-33390> for AdaptiveScheduler.

NO.	State Locality(S)	Minimal TaskManagers (M)	taskmanager.load-balance.mode (T)	Cluster Mode	conflicts items	candidate strategies
1	true	true	default	Application	M & S	• First, narrow down the range of TMs where available slots are located, and

						then allocate these slots according to the state localization principles.
2	true	True	default	Session	same as 1.	same as 1.
3	true	true	slots	Application	M&T; S&T; M&S	1. strategy-1 a. Firstly, do minimal taskmanagers logic b. Select the slot from the task managers with the fewest slots in use, i. match it with the slot that maximizes state in per slot request matching if here're some slots' state-score is equal. 2. Strategy-2 a. Firstly, do minimal taskmanagers logic b. then allocate these slots according to the state localization principles. i. if the state score is equal in the candidate slots, select the suitable slot based on the slots taskmanager load balanced mode.
4	true	True	slots	Session	M&T; S&T; M&S	1. Strategy-1 a. Firstly, don't do minimal taskmanagers logic b. Select the slots from the task managers with the fewest slots in use, i. and match it with the slot that maximizes state in per slot request matching if here're some slots' slots-utilization is equal. 2. Strategy-2

						a. Firstly, don't do minimal taskmanagers logic b. then allocate these slots according to the state localization principles. i. if the state score is equal in the candidate slots, select the suitable slot based on the slots taskmanager load balanced mode.
5	false	true	default	Application	N.A	N.A
6	false	true	default	Session	N.A	N.A
7	false	true	slots	Application	MT & T	• Firstly, do minimal taskmanagers logic • Then, Select the slot from the task managers with the fewest slots in use in per slot request matching
8	false	True	slots	Session	MT & T	• Firstly, don't do minimal taskmanagers logic • Then, Select the slot from the task managers with the fewest slots in use in per slot request matching

Looking forward to your suggestions and comments

!

Thank you very much!