

# Priorisierung

Ende 2020 führen wir ein Bonus-System für den HPC Cluster "Hilbert" ein. Mit einem solchen Bonus ist es möglich, dass Nutzer/innen ihre Jobs selbständig mit einer höheren Priorität versehen können. Priorisierte Jobs werden vom Batchsystem bevorzugt und laufen damit schneller an. Zur Abrechnung werden sogenannte "*PrioPoints*" eingesetzt. Diese werden durch Investitionen, Berufungszusagen und /oder Schulungen im Kontext des HPC-Systems "Hilbert" bereitgestellt. Das dadurch erzielte Kontingent an *PrioPoints* kann jederzeit eingesetzt und für einen bevorzugten Start von Jobs genutzt werden.

Einerseits ermöglicht dieser Mechanismus den Nutzern und Nutzerinnen eigenständig zu entscheiden, welche Jobs ihnen wichtig sind und schnellstmöglich Ergebnisse liefern sollen. Auf der anderen Seite werden dadurch aber keine HPC Ressourcen unnötig freigehalten, sondern alles wird normal genutzt und nur vom Batchsystem entsprechend den Wünschen umgeordnet.

Die Abrechnung sowie die Verwaltung dieser *PrioPoints* geschieht innerhalb von myJAM und kann dort nachvollzogen werden.

- [Verwendung von PrioPoints](#)
- [Informationen im myJAM](#)



## Wartepunkte

Pro Stunde sammelt ein wartender Job automatisch zwei Wartepunkte. Durch Einsatz von *PrioPoints* kann man jedoch Wartezeit einsparen. Wenn man beispielsweise 10 *PrioPoints* einsetzt, dann bekommt der Job weitere 10 Wartepunkte hinzu. Man holt damit quasi die Priorität von 5 Stunden auf.

Die meiste Zeit hat der Top-Job im Cluster einen Wartepunkte-Wert zwischen 200 und 500. Um einen soeben neu eingestellten Job direkt an die oberste Position zu befördern, müsste man also genau so viele Punkte einsetzen - man erwirkt dann, dass der eigene Job die höchste Priorität hat. Die Prioritäten von Jobs kann man sich wie folgt anzeigen lassen.

### Beispiel: Prioritäten im Cluster

```
hpc-login7 ~]$ module load hpc-tools
hpc-login7 ~]$ queue_priority.py | head
6008007.hpc-batch14 = 525.27
6008008.hpc-batch14 = 525.26
6008010.hpc-batch14 = 525.26
6038066[.].hpc-batch14 = 340.57
6038068[.].hpc-batch14 = 340.56
6038069[.].hpc-batch14 = 340.55
6038070[.].hpc-batch14 = 340.54
6038071[.].hpc-batch14 = 340.53
6038072[.].hpc-batch14 = 340.51
6072780.hpc-batch14 = 254.47
```

## Verwendung von PrioPoints

Um die *PrioPoints* einzusetzen, übergibt man **qsub** noch zwei weiteren Parameter namens "bonus" und "anchor" mit entsprechendem Wert.

### Jobscrip Beispiel mit zusätzlichem Bonus-Parameter

```
#!/bin/bash
#PBS -l select=1:ncpus=4:mem=20gb
#PBS -l walltime=1:59:59
#PBS -l bonus=10
#PBS -l anchor=Test
#PBS -N ...
```



Hinweise: Dieses Bonus-System ist nur für Jobs in normalen Queues anwendbar und nicht in Sonderqueues wie **long\_ayor**. Einmal eingesetzte *PrioPoints* können nicht wieder zurückgezogen oder erstattet werden. Bricht ein Job aus welchem Grund auch immer ab oder wird er vom HPC Team, beispielsweise bei unsachgemäßer Nutzung der Ressourcen, abgebrochen, dann werden die eingesetzten *PrioPoints* dennoch vom Kontingent abgezogen.

## Informationen im myJAM

Zu jedem Job wird angezeigt, ob dieser PrioPoints verwendet hat und aus welchem Anchor diese abgerechnet wurden.

PrioPoint Usage	
Anchor	Rehs_Test
Budget-Name	Verbrauch
Neuer Test	400

Darüber hinaus wird bei jedem Projekt angezeigt, welche Anchor diesem zugeordnet sind und verwendet werden können.

✚ PrioPoint Anchors
Rehs_Test

In den Anchor-Details wird angezeigt, welche Budgets dort zugeordnet sind und wie viel davon verbraucht wurde.

✚ Budgets		
Date-Range	Points used	Job count
17.09.2021 - 29.01.2022 Für Publikation	0 / 1500	0
28.01.2022 - 10.03.2022 Neuer Test	8000 / 20000	20