

Big-Data-Technologien

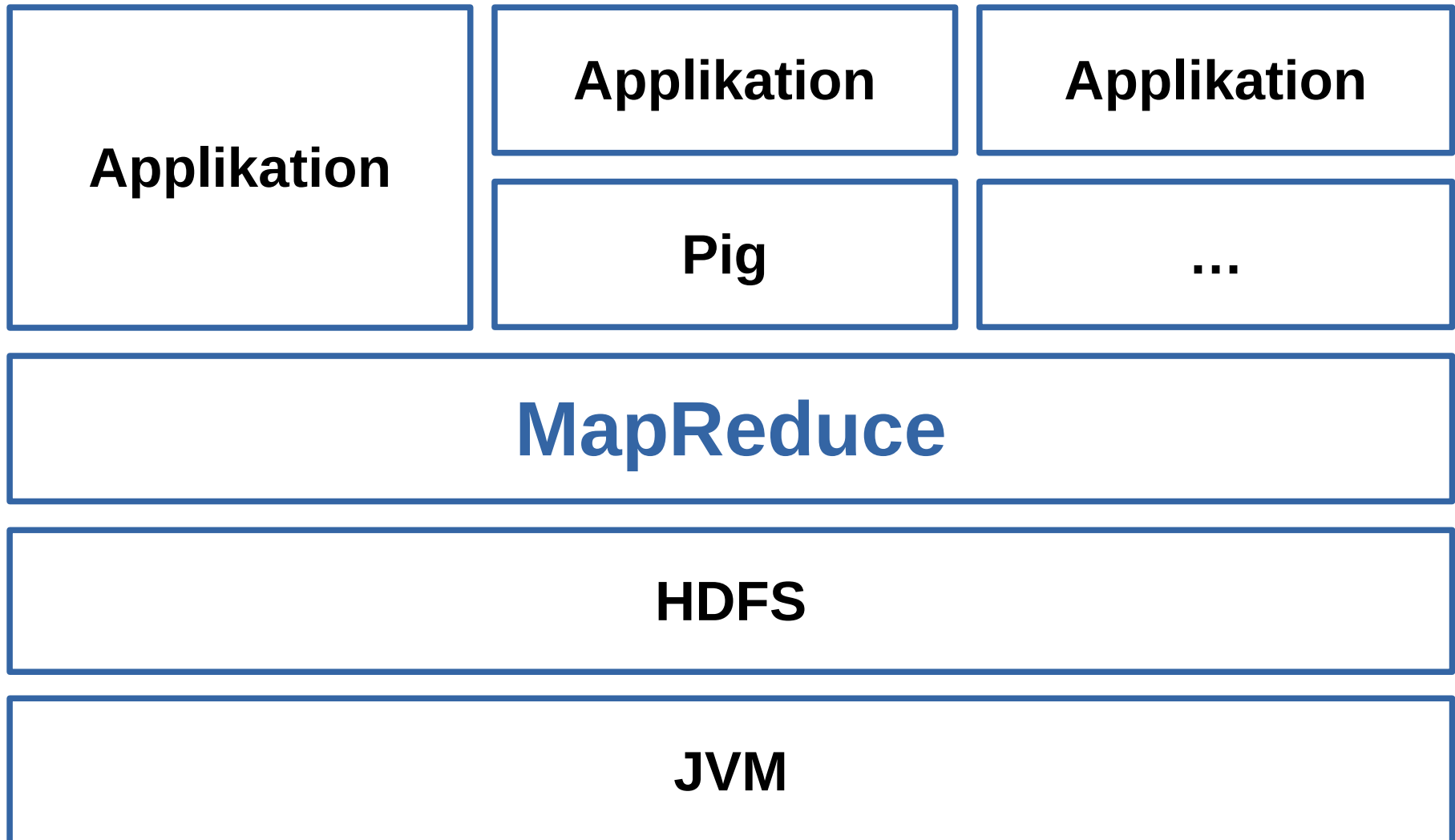
Kapitel 5: Hadoop – YARN

Hochschule Trier
Prof. Dr. Christoph Schmitz

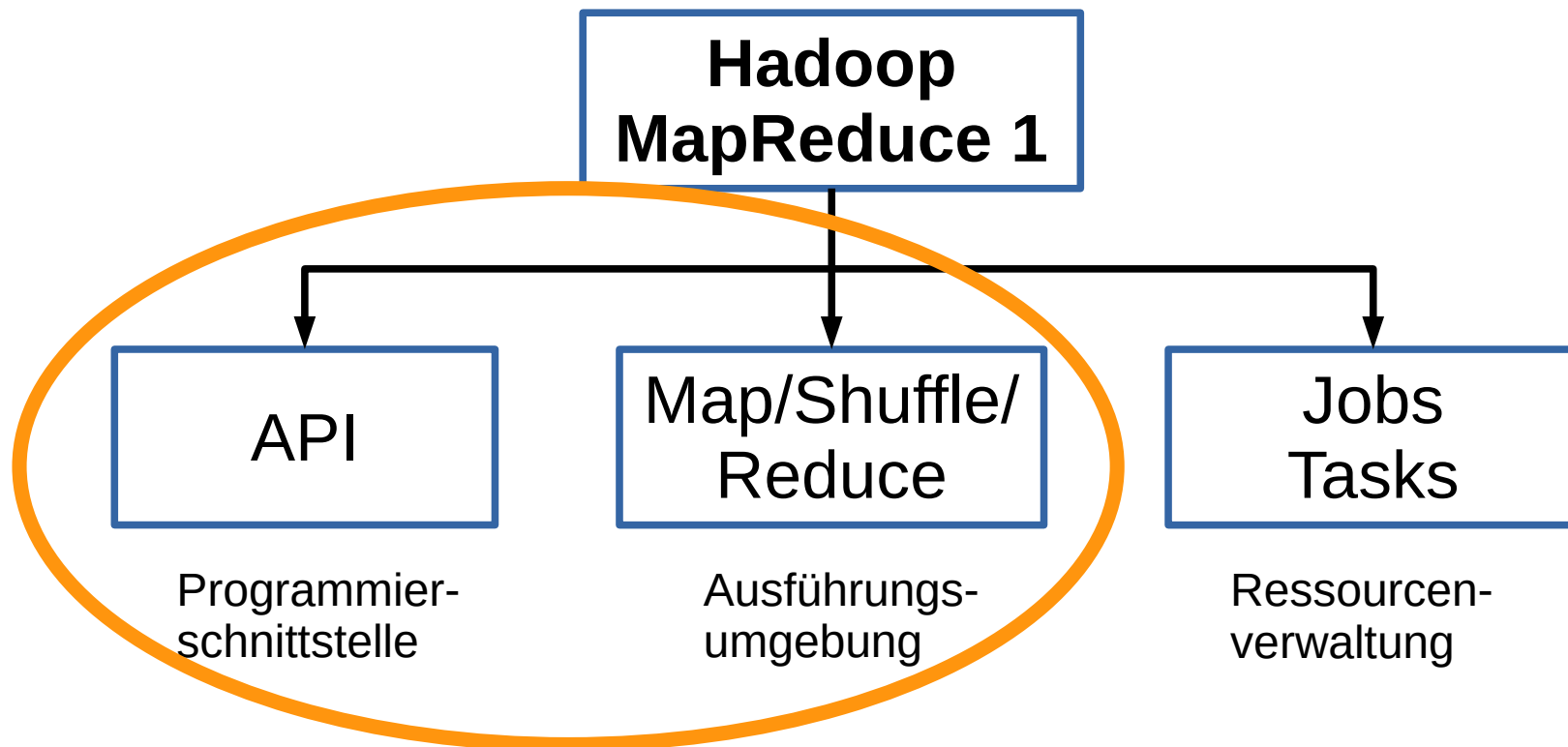
Überblick

- Warum YARN?
- Zentrale Konzepte
- Ablauf einer Applikation in YARN
- Scheduling
- Sicherheit

Architektur von Hadoop (vor 2012)



MapReduce: drei Aufgaben



Warum YARN?

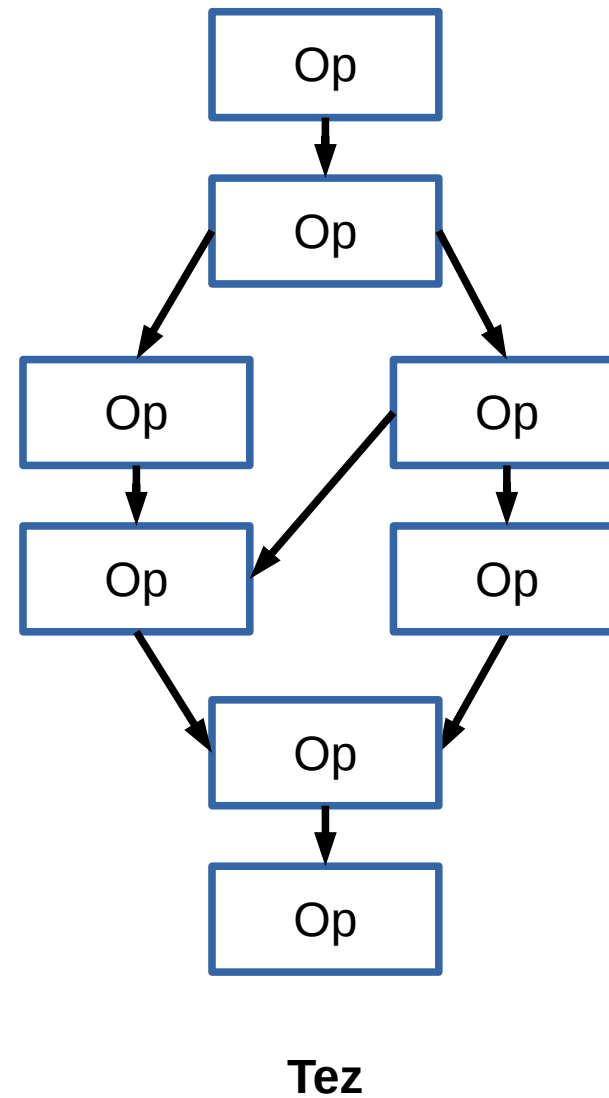
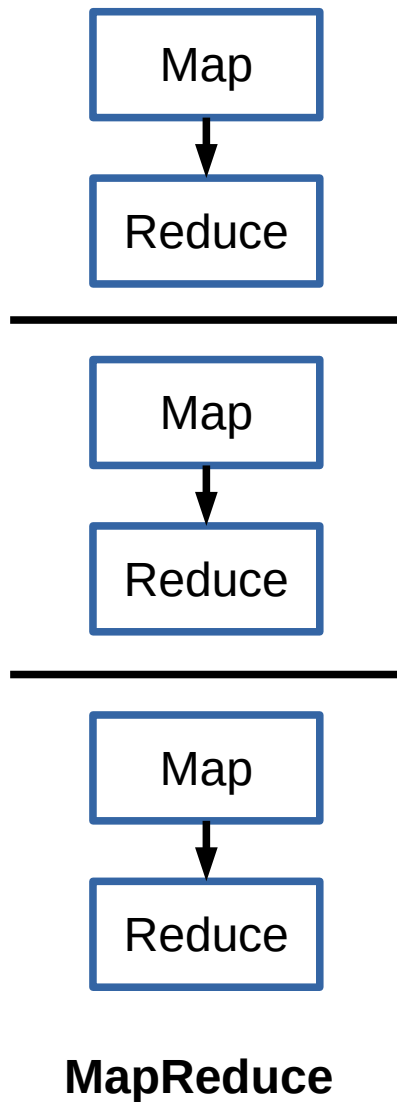
- Hadoop-Cluster anderweitig nutzen
- Hochverfügbarkeit
- Skalierbarkeit (→ 10.000 Knoten!)
- Auslastung erhöhen
- HDFS beibehalten

➔ **YARN: Yet Another Resource Negotiator**

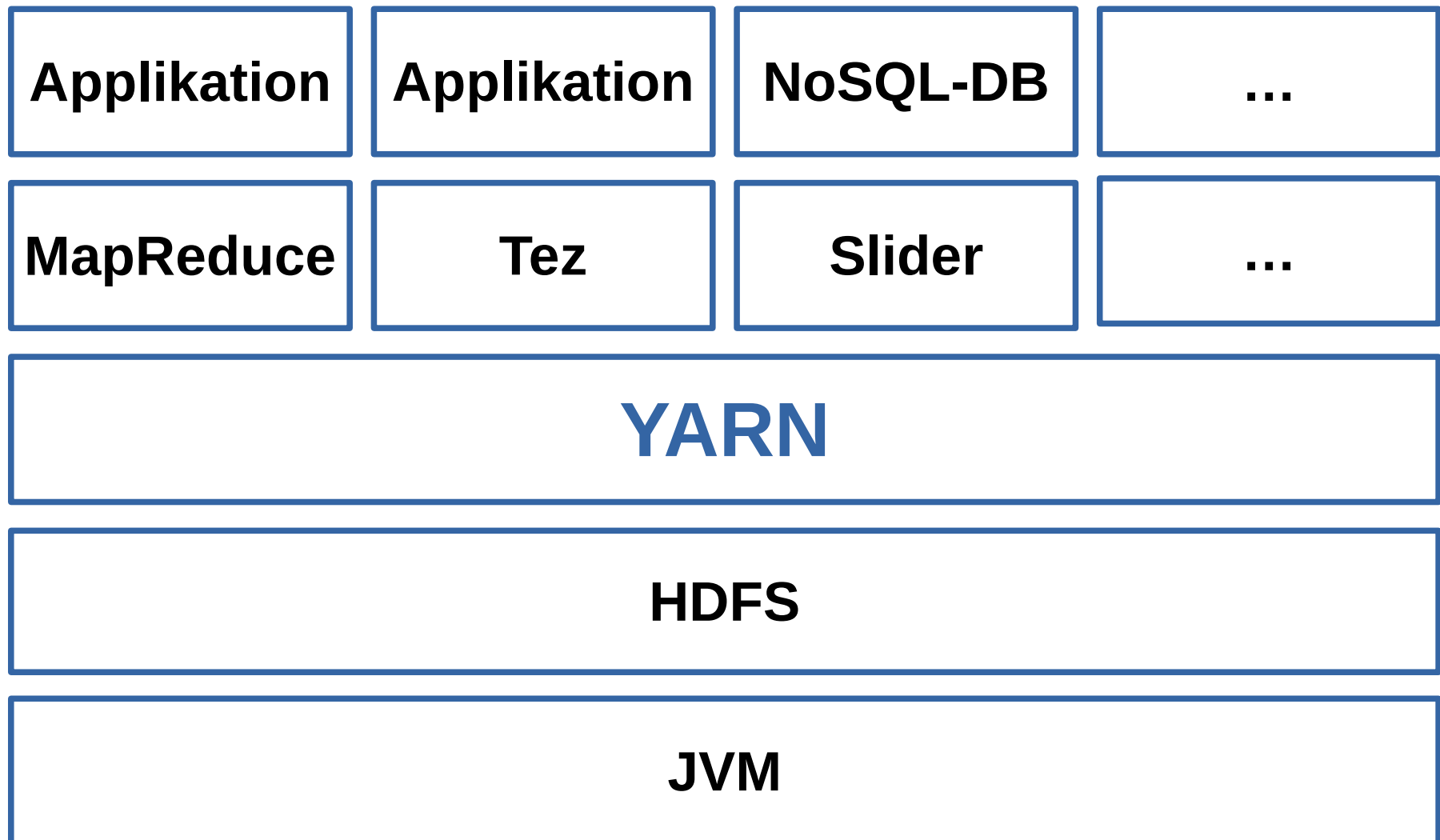
Alternative Nutzung des Hadoop-Clusters

- Effizientere/komfortablere Programmiermodelle
 - z. B. Datenfluss-Graphen → **Tez**, **Spark**
- Langlaufende Dienste
 - z. B. Datenbank-Server → **Slider**
- Stream-Verarbeitung
 - z. B. **Spark Streaming**, **Flink**

Beispiel: Tez



Hadoop-Architektur mit YARN



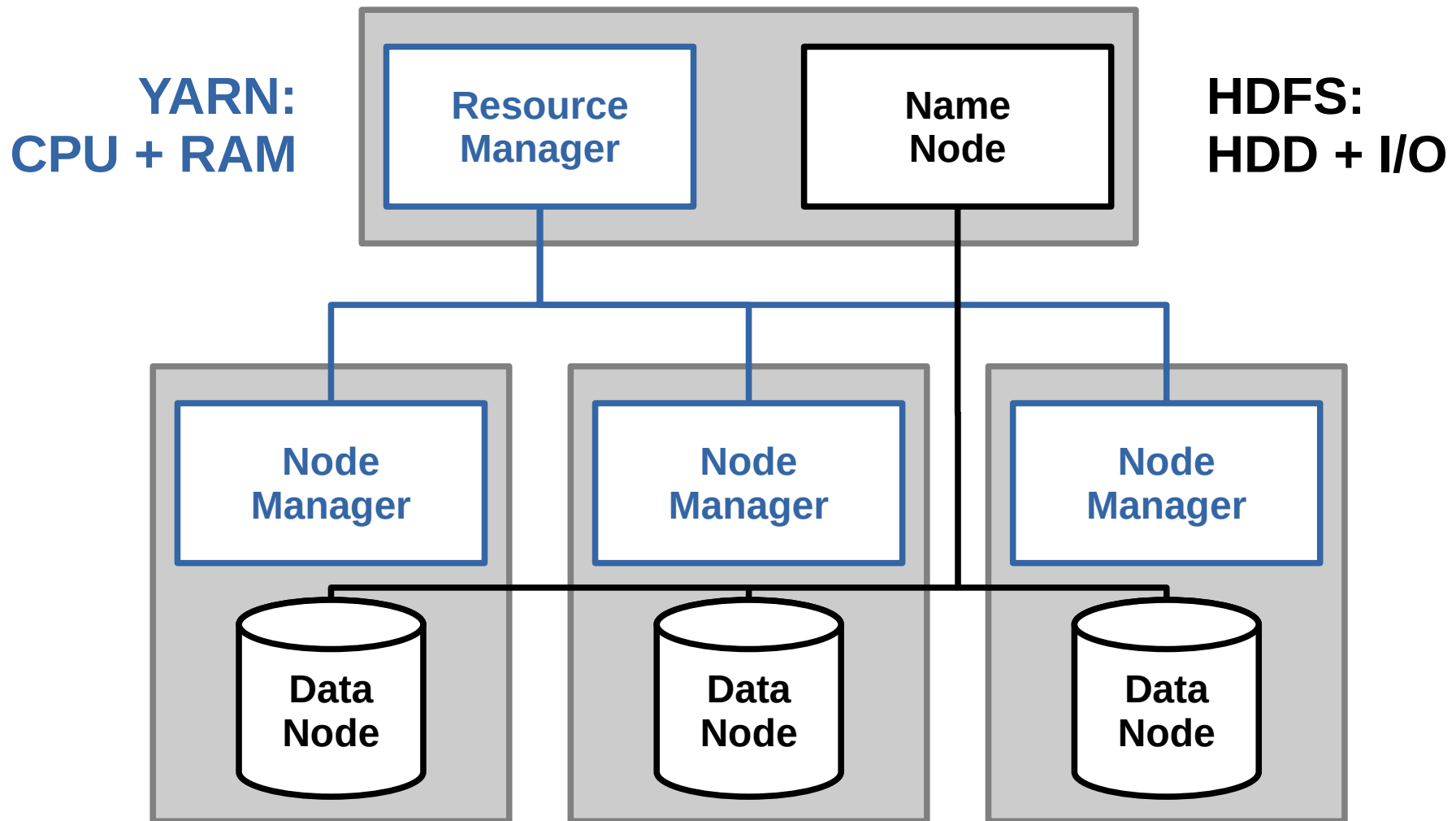
Was verwaltet YARN?

- Zuweisung von
 - **Rechenleistung** (CPU-Kernen)
 - **Hauptspeicher**
 - **Rechenzeit** (Scheduling)
 - Kann **Datenlokalität** berücksichtigen
- Applikationen laufen in **Containern**
- GUI (Monitoring)

Zentrale Konzepte in YARN

- **Container**
 - Zugewiesene Ressourcen (Kerne + RAM) auf einem Knoten
 - Ausführungsumgebung für Applikationen
- **NodeManager**
 - Verwaltet einen Knoten
- **ResourceManager**
 - Zentrale Verwaltung des gesamten Clusters
- **ApplicationMaster**
 - Steuerung der Ressourcen einer Applikation

Zwei parallele Subsysteme



Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved
0 B	25 GB	0 B	0	10	0

Cluster Metrics

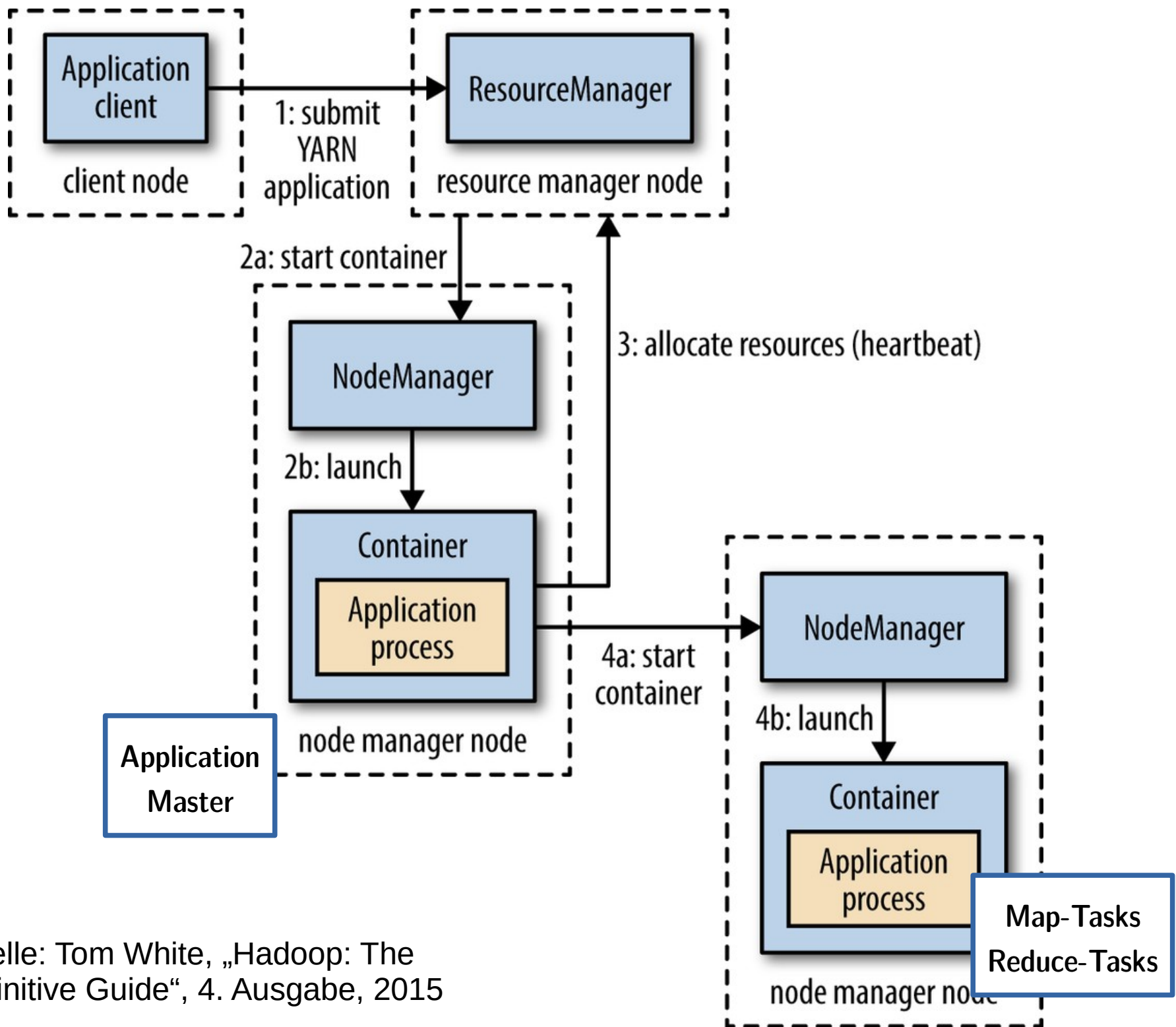
Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
2	0	0	2	0	0 B	25 GB	0 B	0	10	0	5	0	0	0	0

Scheduler Metrics

Node HTTP	Containers	Mem Used	Mem Avail	VCores	VCores
nfbdt01.fh-rier.de:8042	0	0 B	5 GB	0	2
nfbdt02.fh-rier.de:8042	0	0 B	5 GB	0	2
nfbdt03.fh-rier.de:8042	0	0 B	5 GB	0	2
nfbdt04.fh-rier.de:8042	0	0 B	5 GB	0	2
nfbdt05.fh-rier.de:8042	0	0 B	5 GB	0	2

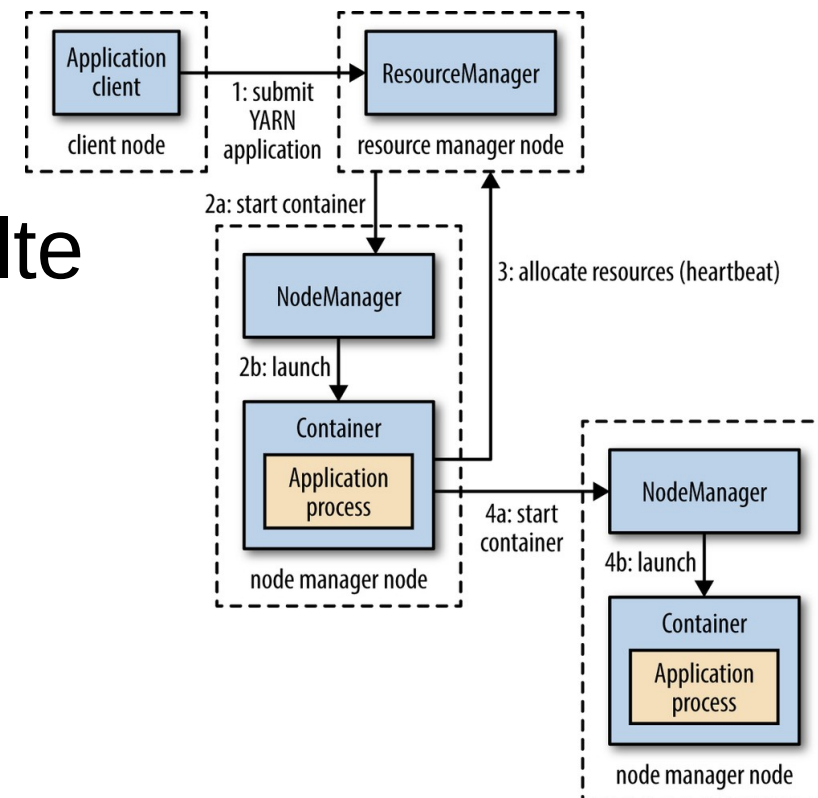
YARN-GUI

Show 20 entries			Application Type	Application ID	Application Priority	StartTime	FinishTime	State	FinalStatus
ID	User	Name							
application_1486467571635_0002	hive	HIVE-7d40aa4c-3c8e-46ec-866f-1dfc041e8385	TEZ	application_1486467571635_0002	0	Thu Feb 9 09:44:31 +0100 2017	Thu Feb 9 09:56:50 +0100 2017	FINISHED	SUCCESS
application_1486467571635_0001	bigdata200	PigLatin:DefaultJobName	MAPREDUCE	application_1486467571635_0001	0	Thu Feb 9 09:31:17 +0100 2017	Thu Feb 9 09:31:57 +0100 2017	FINISHED	SUCCESS
application_1486461173640_0001	bigdata200	Spark shell		application_1486461173640_0001	0	Tue Feb 7 12:15:46 +0100 2017	Tue Feb 7 12:30:27 +0100 2017	FINISHED	SUCCESS
application_1485435061147_0006	bigdata200	hadoop-archives-2.7.3.2.5.3.0-37.jar	SPARK	application_1485435061147_0006	0	Thu Feb 2 13:16:54 +0100 2017	Thu Feb 2 13:17:20 +0100 2017	FINISHED	SUCCESS
application_1485435061147_0005	bigdata200	HIVE-f365eb8e-1718-4a6d-91f0-262ee9804c55	MAPREDUCE	application_1485435061147_0005	0	Thu Jan 26 14:14:58 +0100 2017	Thu Jan 26 14:16:22 +0100 2017	FINISHED	SUCCESS
application_1485435061147_0004	bigdata200	select count(*) from foo group by bar % 5(Stage-1)	TEZ	application_1485435061147_0004	0	Thu Jan 26 14:14:00 +0100 2017	Thu Jan 26 14:14:25 +0100 2017	FINISHED	SUCCESS
			MAPREDUCE						



Ablauf im Detail

- Client-Applikation schickt Code an **ResourceManager**
- ResourceManager startet **Application Master** in einem Container
- Application Master startet **weitere Container** für verteilte Berechnung



Was lebt wo?

ResourceManager NodeManager	Application Master	Client-Applikation	NameNode DataNode
Verwaltung von CPUs	Application Framework	Geschäftslogik	Verwaltung von Sekundärspeicher (HDD)
Verwaltung von RAM	z. B. MapReduce	z. B. Mapper, Reducer	
Management der Container			
Scheduling			Rechte und Quotas
Hochverfügbarkeit			Hochverfügbarkeit

Brauche ich eigentlich ein Application Framework?

- Lässt sich eine Applikation auch direkt auf YARN aufsetzen?

Ja, aber das will man nicht.

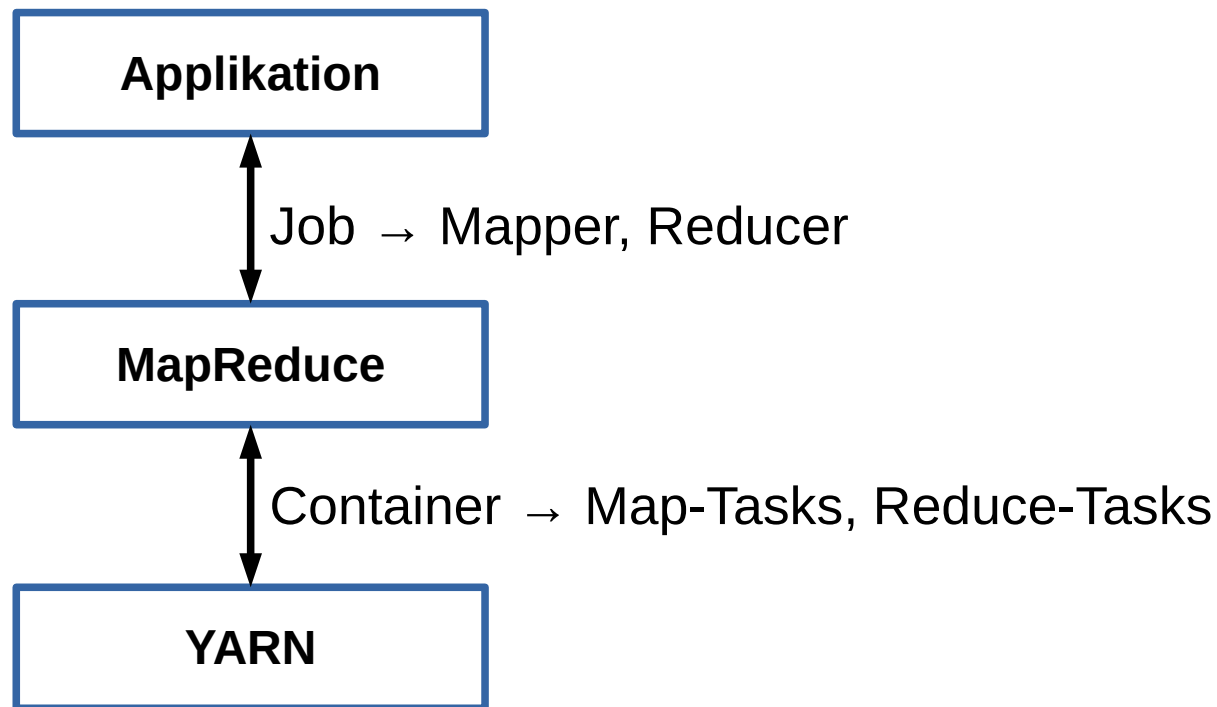
→ **Apache Twill**

Lebenszyklus einer YARN-Applikation

- Eine Applikation **pro User-Job**
 - Beispiel: MapReduce
- Eine Applikation **pro Datenfluss/Session**
 - Beispiel: Spark, Tez
- **Langlaufende** Applikationen
 - Beispiel: Datenbank-Server

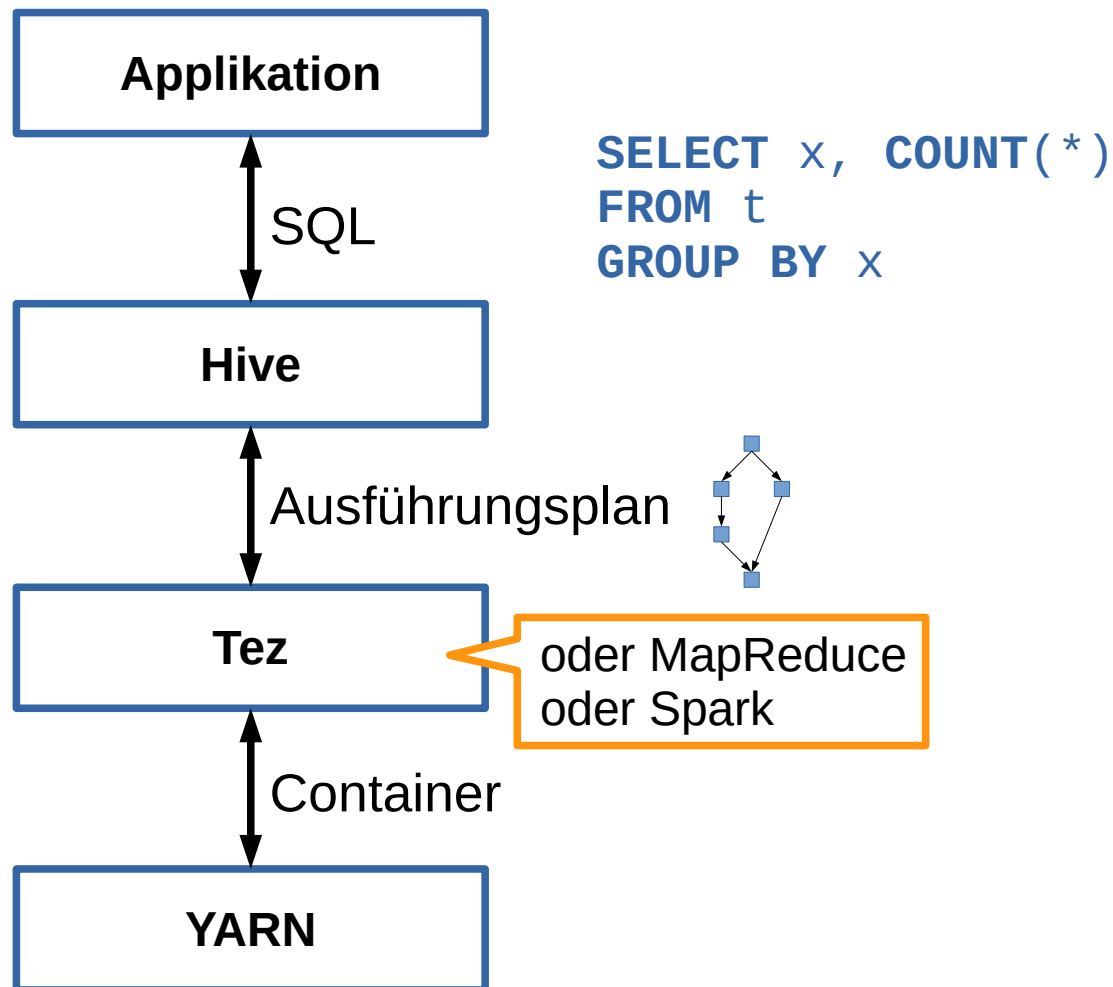
Typische Verwendungen von YARN

- Beispiel: MapReduce



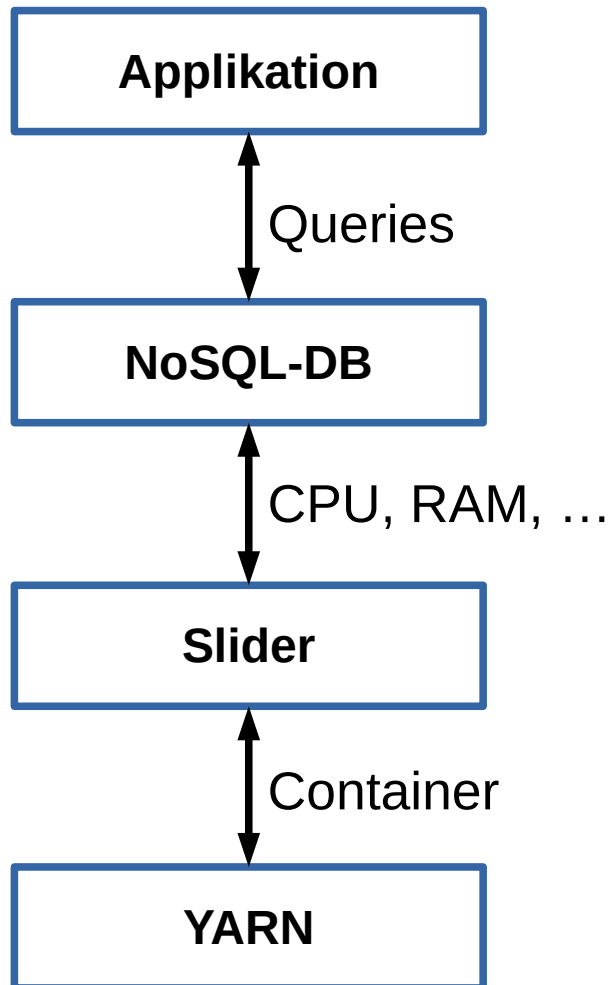
Typische Verwendungen von YARN

- Beispiel: SQL mit Hive



Typische Verwendungen von YARN

- Beispiel: NoSQL-Datenbank



Scheduling

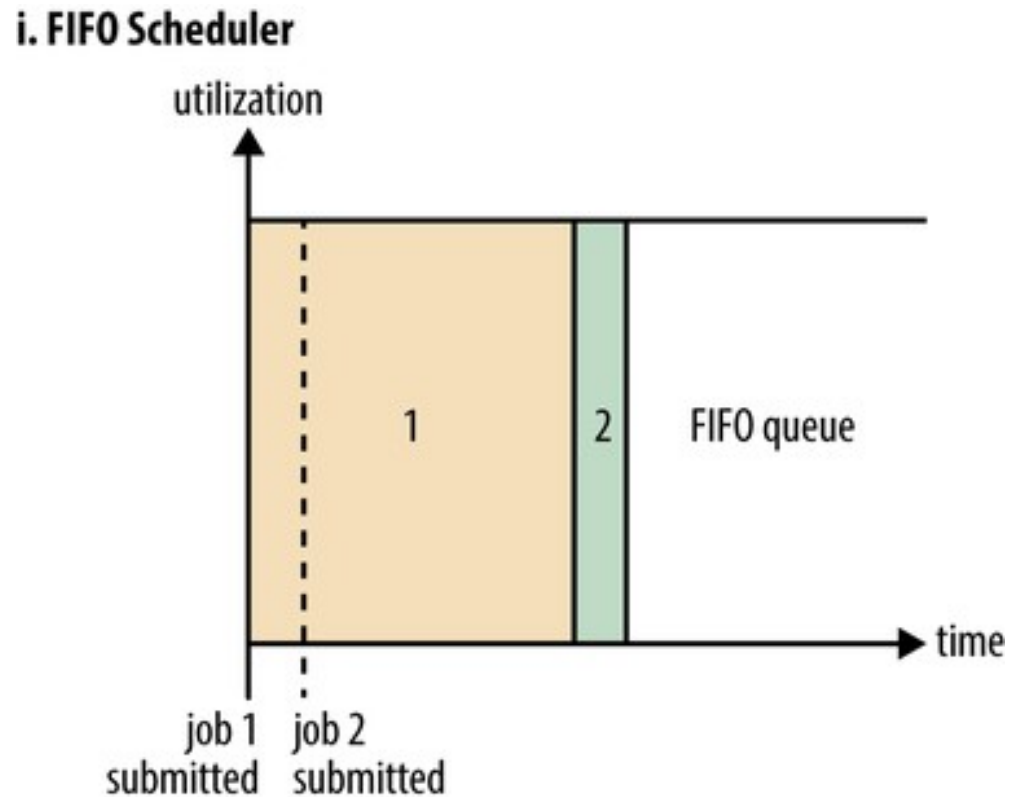
- Wer darf wann wieviel rechnen?
- Widerstrebende Ziele
 - **Fairness**
 - **Auslastung**
 - **Zieltermine**
 - **Prioritäten**
 - **Verhungern vermeiden**

Scheduler in YARN

- FIFO Scheduler
- Fair Scheduler
- Capacity Scheduler

FIFO Scheduler

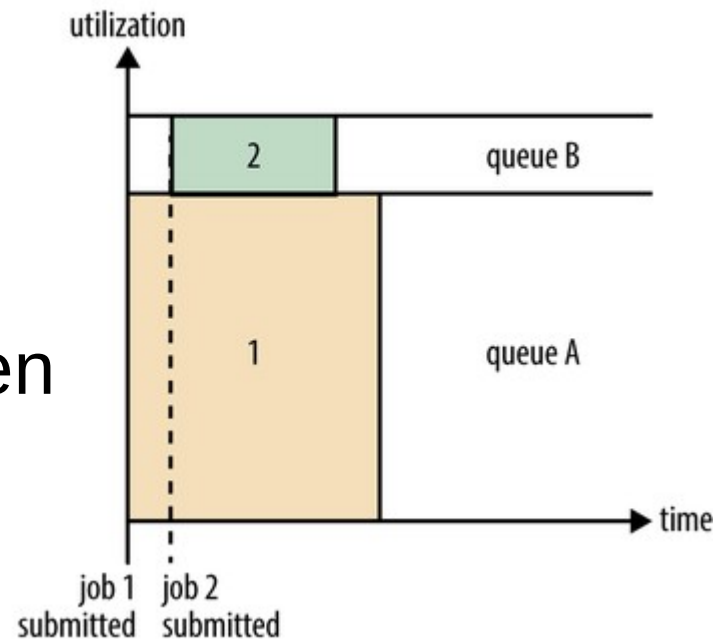
- Bearbeite Anfragen in der Reihenfolge des Einreichens
- Pro:
 - Leicht verständlich
 - Auslastung
- Kontra:
 - Fairness
 - Prioritäten
 - Langläufer



Capacity Scheduler

- Unterschiedliche Warteschlangen (Queues)
 - z. B. für Organisationseinheiten
 - Hierarchie
- Kapazität pro Warteschlange
 - Maximum pro Job
 - Kann in Summe überschritten werden (→ Auslastung)

ii. Capacity Scheduler



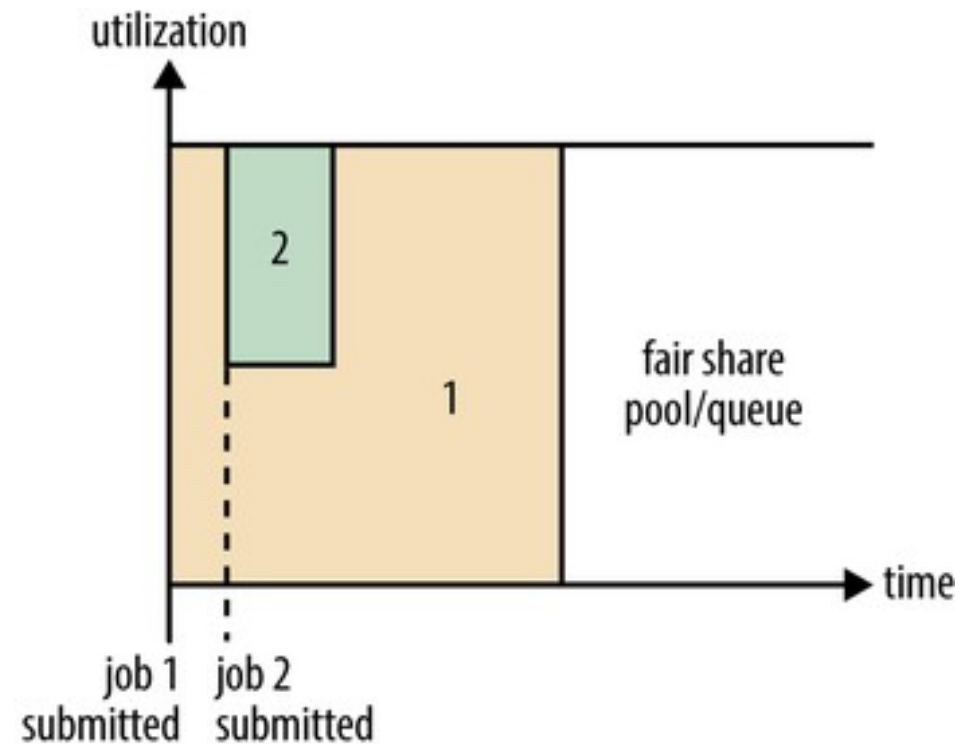
Capacity Scheduler

- Pro:
 - Prioritäten
 - Fairness
- Kontra:
 - Konfigurationsaufwand
 - Auslastung vs. Fairness

Fair Scheduler

- Gleiche Anteile für jeden Job
- Gewichtete Queues
- Pro:
 - Fairness
 - Auslastung
- Kontra:
 - Komplexität/
Konfigurationsaufwand

iii. Fair Scheduler



Security

- Wie HDFS: Authentifizierung über Kerberos
- Autorisierung
 - **Service-Level Authorization:**
Wer darf überhaupt mit den Diensten reden?
 - **ACLs (Access Control Lists) im Scheduler:**
Wer darf in welche Queue Jobs einstellen?
- **Frontends** dafür in den Distributionen
 - Ranger
 - Cloudera Manager
 - ...

Service Manager

bdt_yam Policies

Edit Policy

Edit Policy

Policy Details :

Policy Type **Access**Policy ID **3**

Policy Name * all - queue

enabled

Queue * % *

Recursive **ON**Audit Logging **YES**

Description Policy for all - queue

Allow Conditions :

Select Group	Select User	Permissions	Delegate Admin
<div>Select Group</div>	<div>% yam % rangerlookup % ambari-qa</div>	<div>submit-app admin-queue</div>	<div><input checked="" type="checkbox"/> <input type="checkbox"/></div>

+

Save

Cancel

Delete