XtreemOS

Enabling Linux
for the Grid

Grid Checkpointing

John Mehnert-Spahn

Heinrich-Heine University Duesseldorf, Germany

XtreemOS Summer School, Günzburg, Germany, 2010

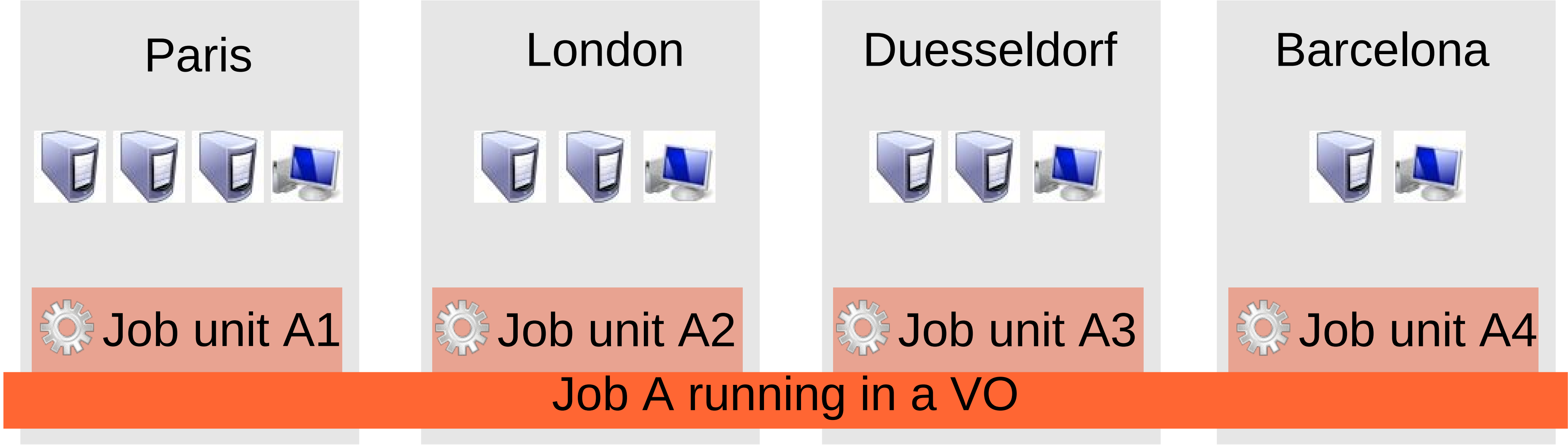
Information Society
Technologies

XtreemOS IP project
is funded by the European Commission under contract IST-FP6-033576

Overview

- Checkpointing
- XtreemGCP
- Communication channel checkpointing with heterogeneous checkpointers
- (Adaptive Checkpointing – incremental grid cp)

Grid Jobs



Paris

London

Duesseldorf

Barcelona

Job unit A1

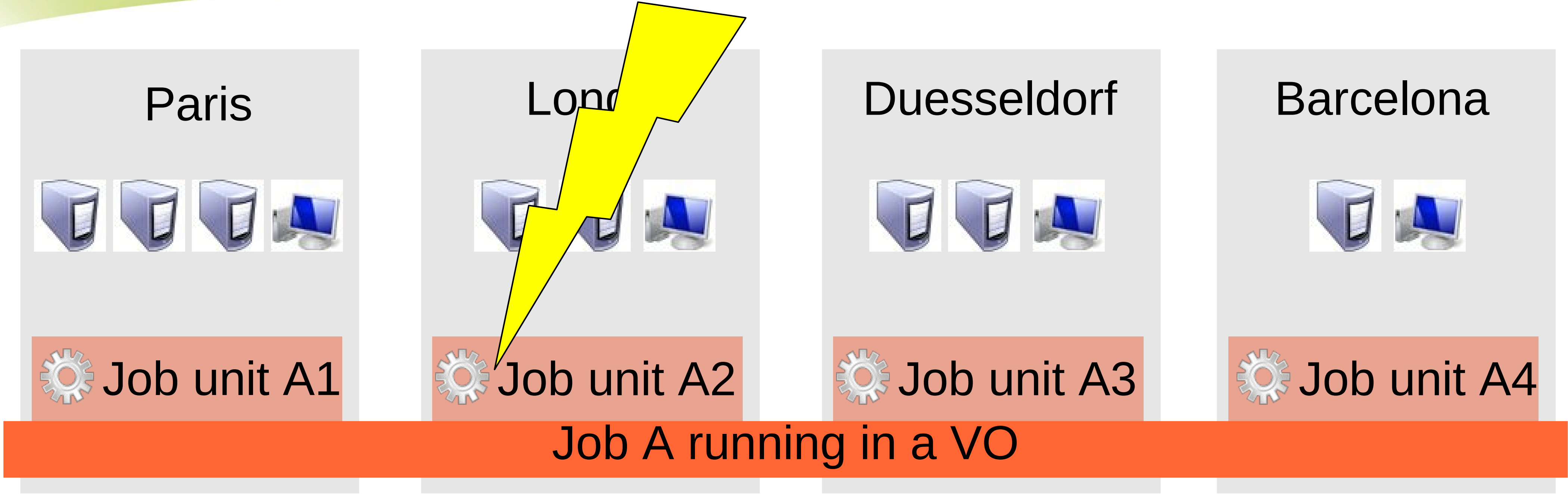
Job unit A2

Job unit A3

Job unit A4

Job A running in a VO

Faults



Fault tolerance needed

Fault tolerance

- Replication
- Forward error recovery
- Backward error recovery

Checkpointing & Restart

- **Checkpointing: The application state is saved periodically to stable storage.**

- **Restart: The application gets reestablished from a recent checkpoint. Thus, no fall back to the initial state will occur.**

Checkpointing & Restart

- Checkpointing: Saving periodically the state of the application in stable storage

- Restart: In case of a fault we can restart from a checkpoint and do not fall back to the initial state

- Challenges:
  - Trade-off between costs during fault-free execution and costs at recovery
  - Size of the distributed state may be very large
  - Checkpointing images must be replicated
  - Heterogeneity of checkpointer packages

XtreemOS IP project
is funded by the European Commission under contract IST-FP6-033576

Many Checkpointers exist

DMTCP & MTCP

BLCR

KMU

TICK

MCR

CHPOX

DCR

zap

CRAK

CLIP

libckpt

tmPVM

Dynamite

VMWare player

Linux-native

SCore

OpenVZ

LinuxSSI

Ckpt

LAM/MPI&BLCR

VMADump

CoCheck

UCLiK

Epckpt

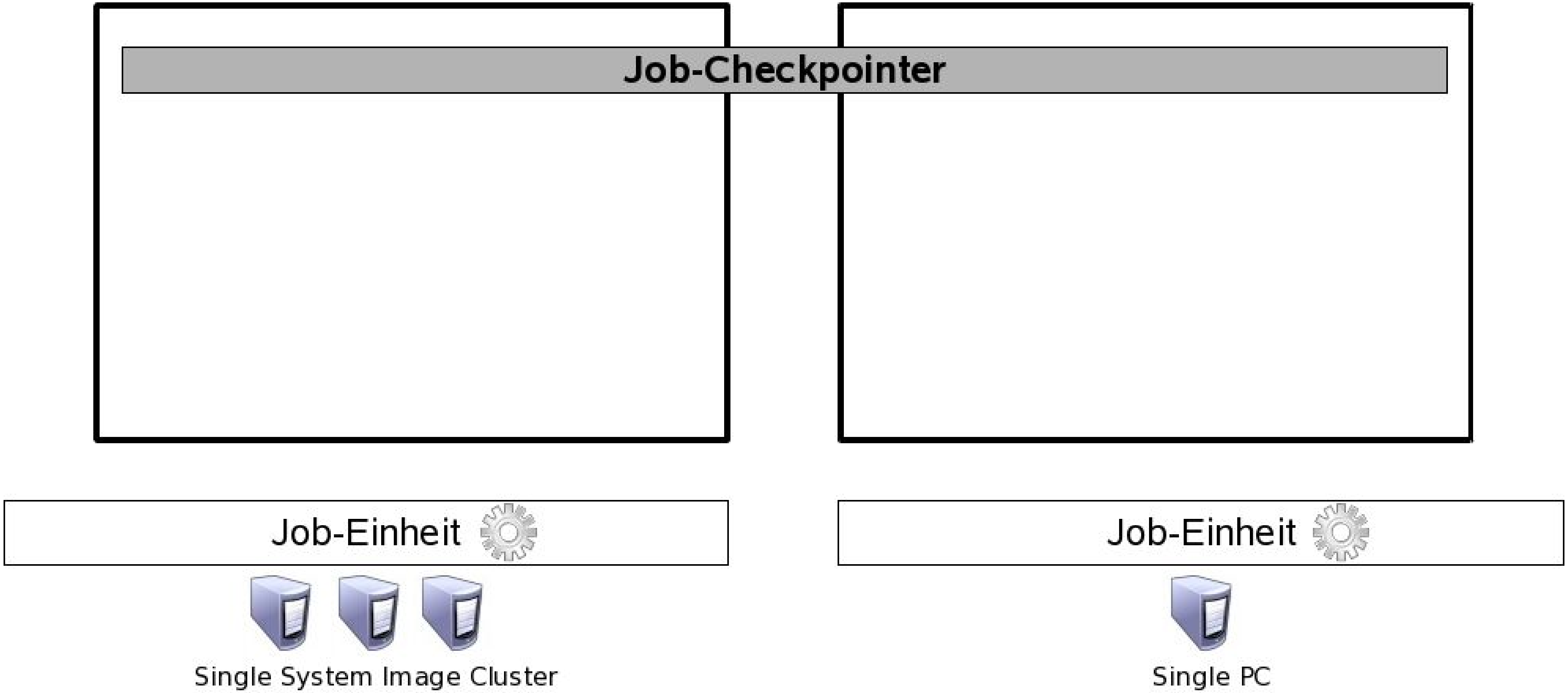
Condor

XtreemGCP checkpointing service

XtreemGCP

- A grid service integrated within AEM implementing **job migration** and **job fault tolerance** for grid jobs
- Integrates existing checkpointer packages
- Supports transparent and application-level checkpointing
- Security

Grid-Checkpointing Architecture



Job-Checkpointer

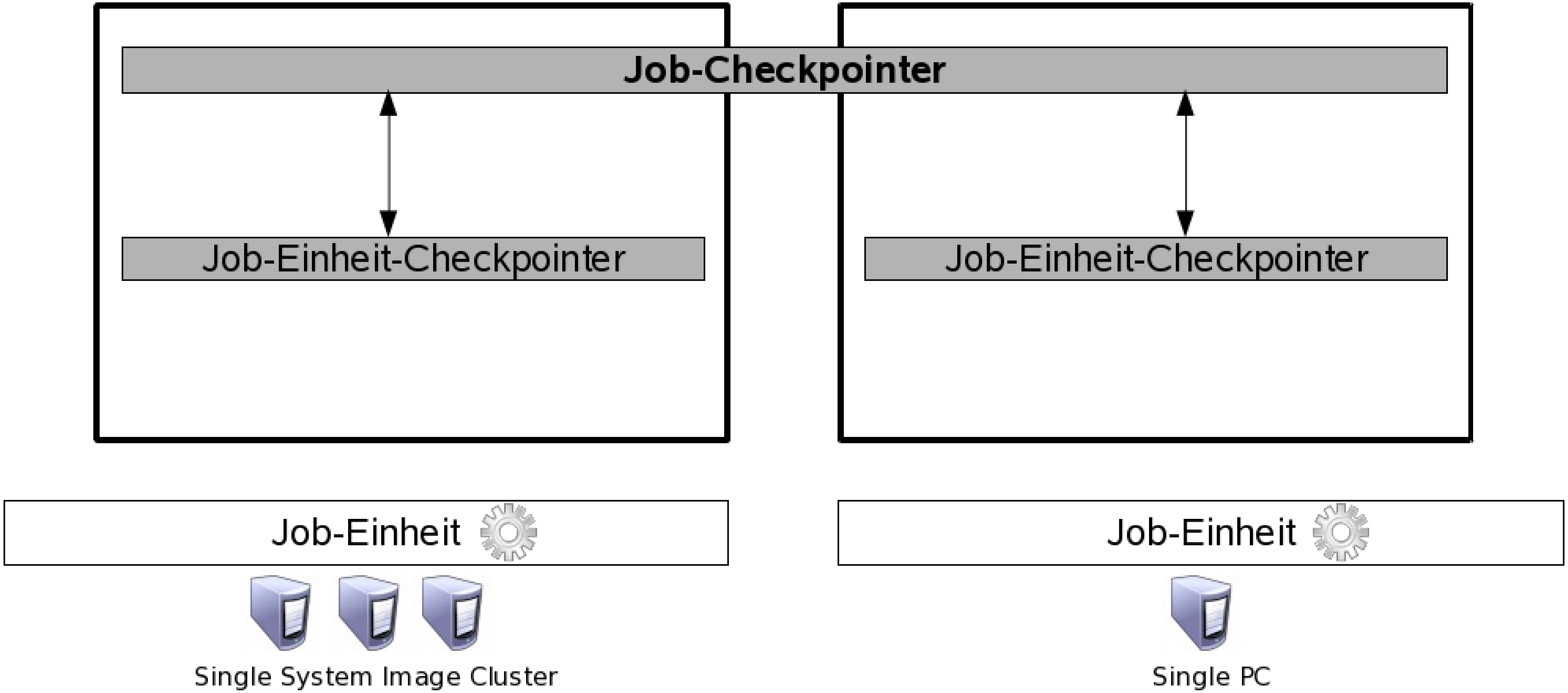
Job-Einheit

Job-Einheit

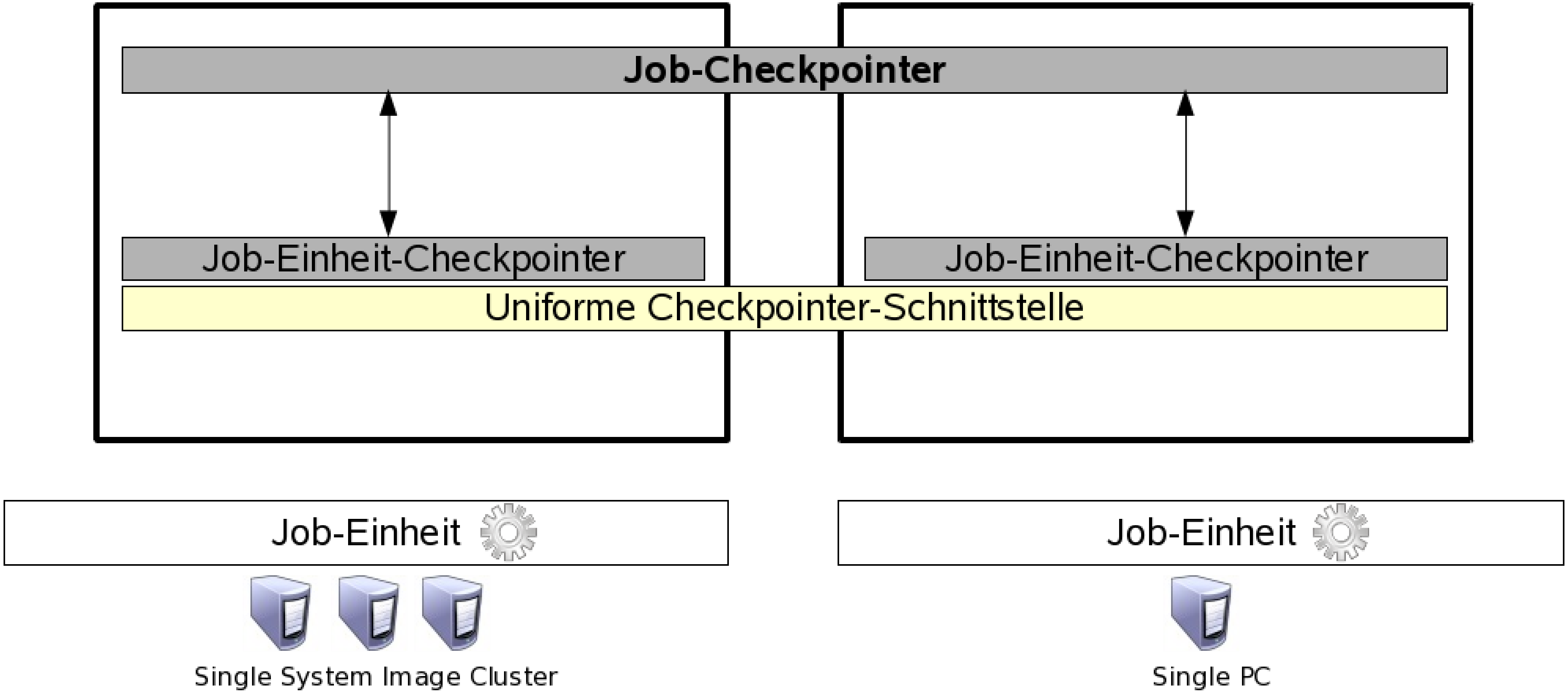
Single System Image Cluster

Single PC

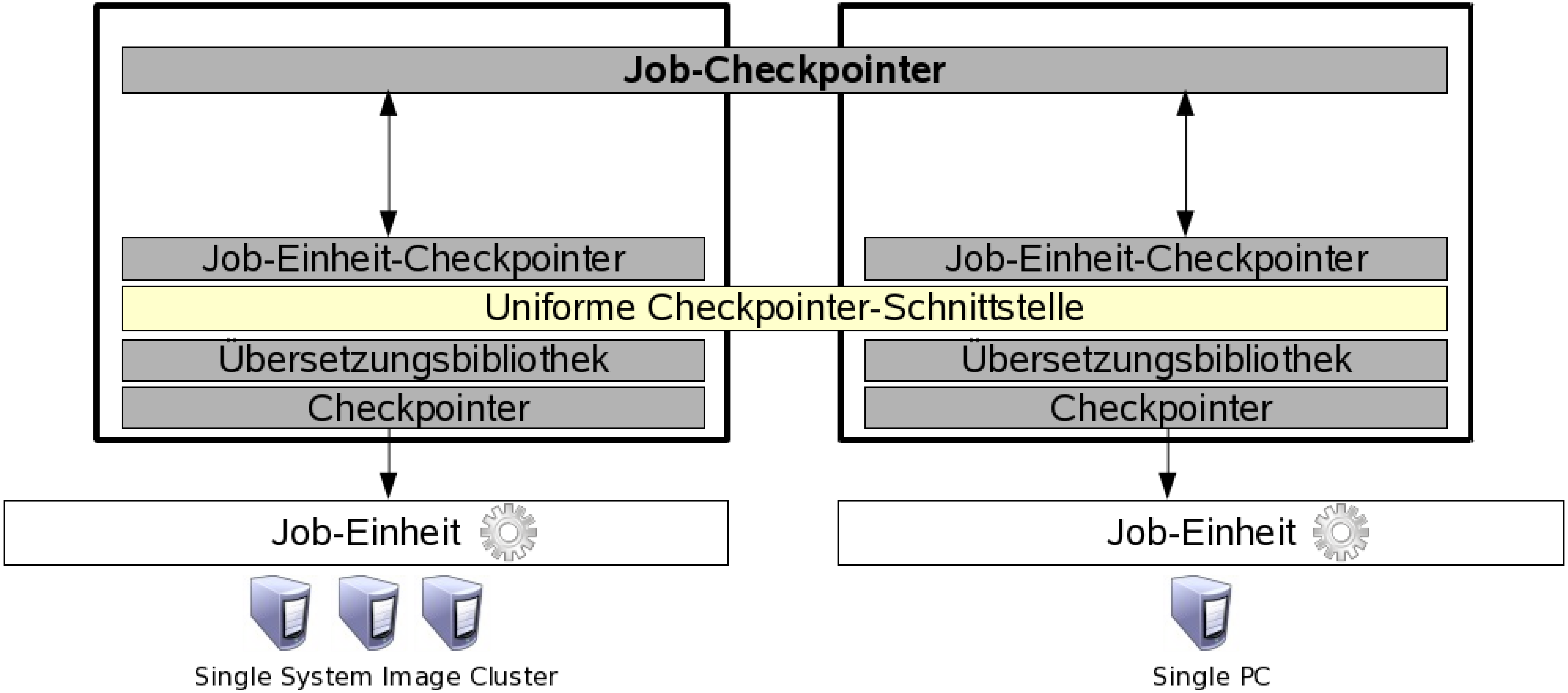
Grid-Checkpointing Architecture



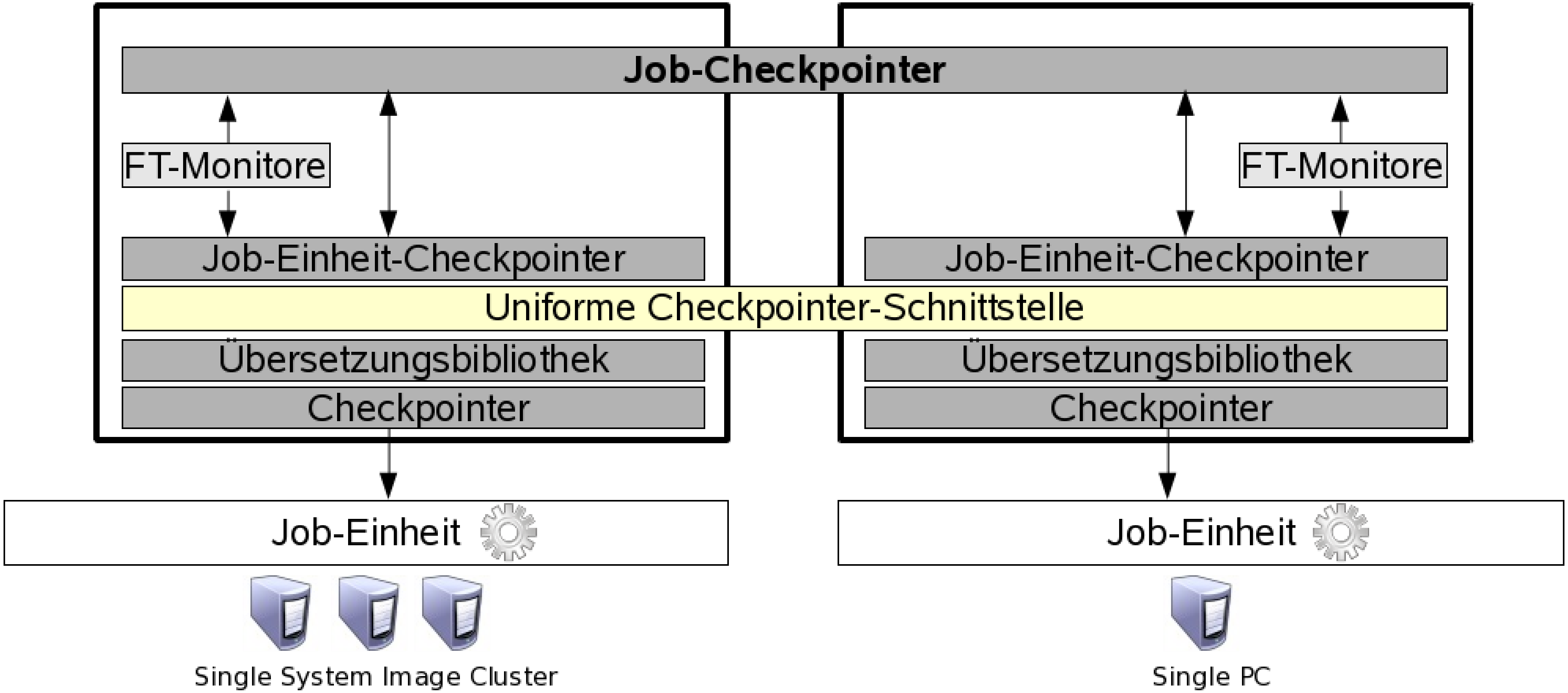
Grid-Checkpointing Architecture



Grid-Checkpointing Architecture



Grid-Checkpointing Architecture



Job-Checkpointer

FT-Monitore

Job-Einheit-Checkpointer

Uniforme Checkpointer-Schnittstelle

Übersetzungsbibliothek

Checkpointer

FT-Monitore

Job-Einheit-Checkpointer

Übersetzungsbibliothek

Checkpointer

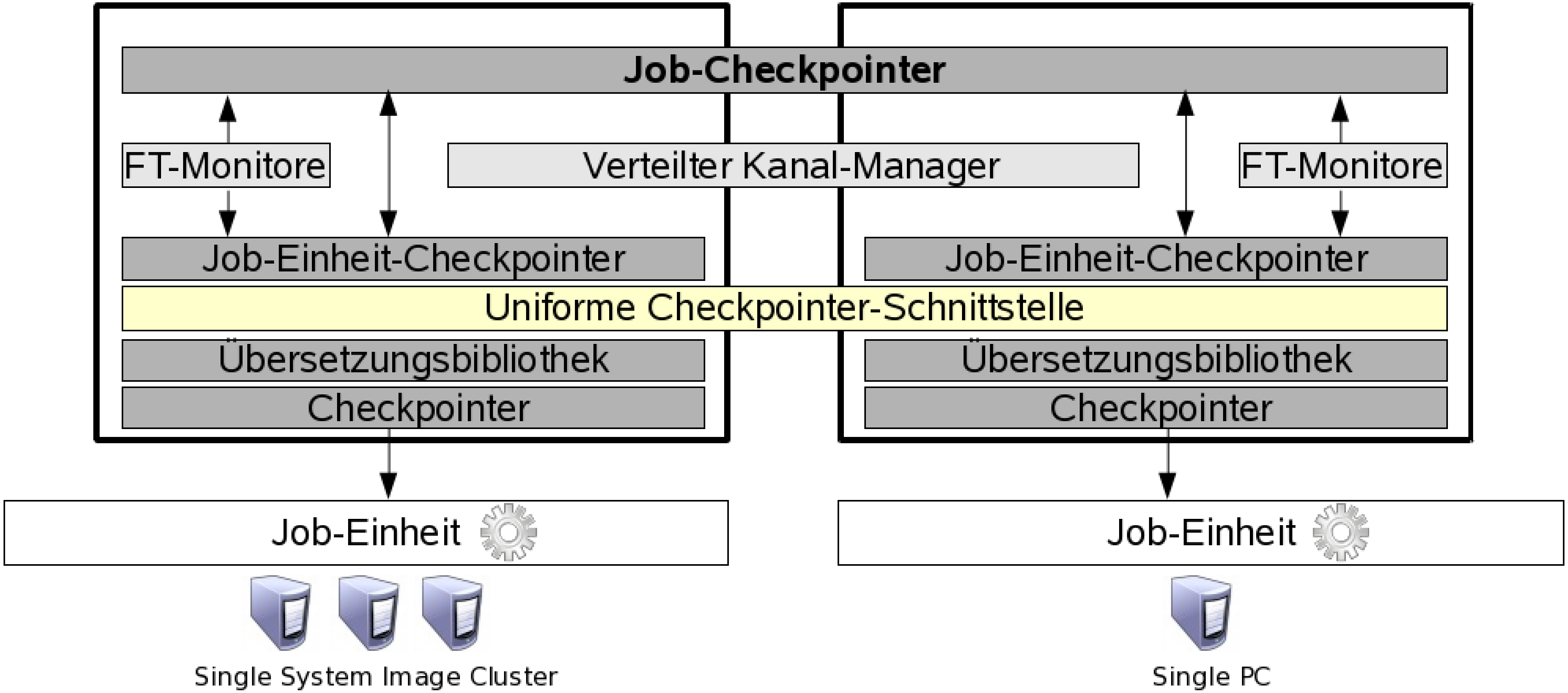
Job-Einheit

Job-Einheit

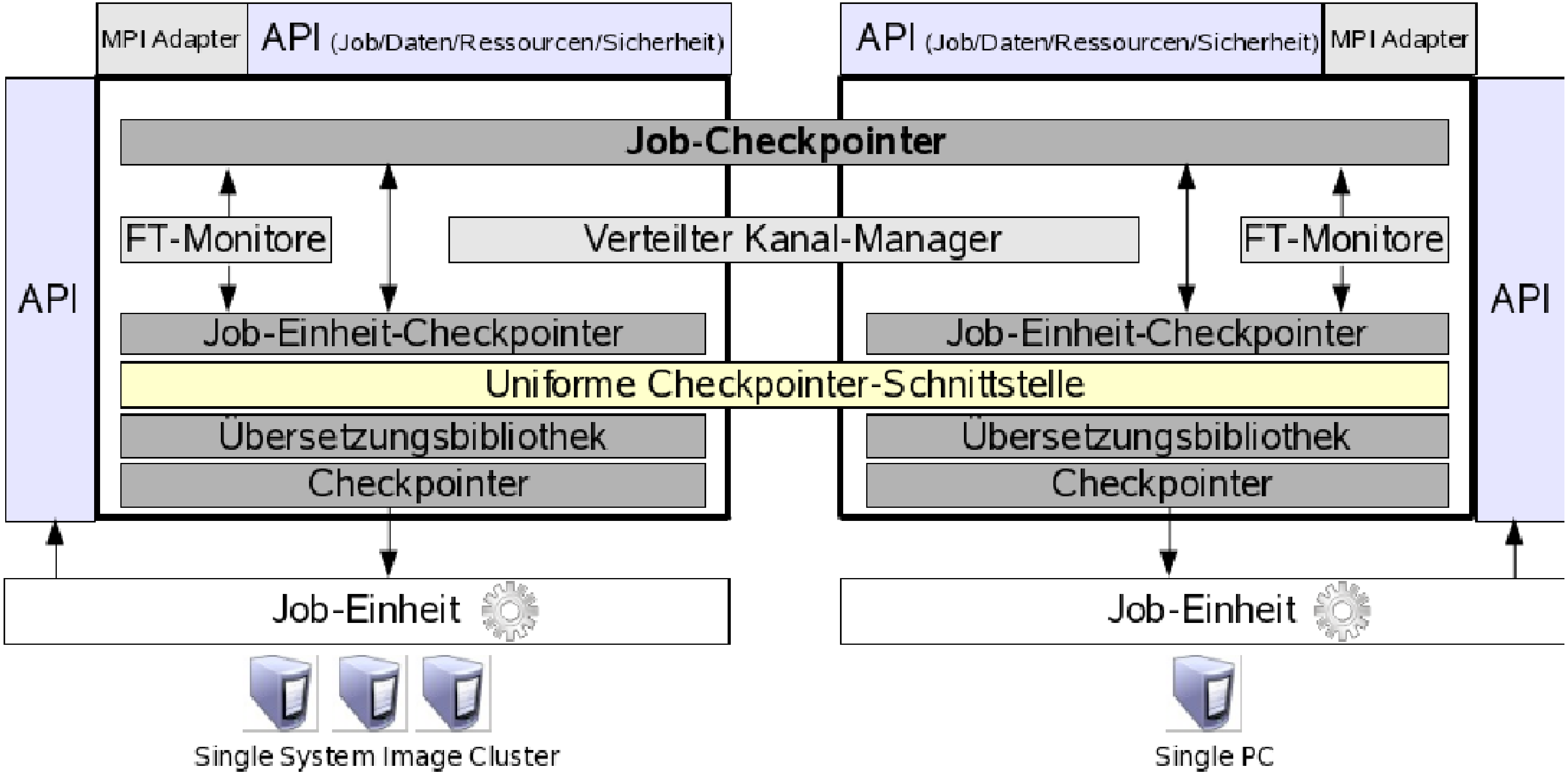
Single System Image Cluster

Single PC

Grid-Checkpointing Architecture



Grid-Checkpointing Architecture



Uniform Checkpointer Interface

- **Uniform access to different checkpointer packages**
  → implemented by a translib (shared library)
- **Translations**
  - **function signatures**
  - **job-to-Linux process group**
  - **grid user id-to-local user id**
  - **callback management**
  - **checkpoint image dependencies**
  - **checkpointer-to-checkpointer**
  - **application-checkpointer-compatibility**

Uniform Checkpointer Interface

- **To which extent must existing checkpointers be adapted to support various checkpointing protocols?**

- **We need the following sequences**
  - **Stop**
  - **Checkpoint**
  - **Resume_cp**

  Checkpoint

  - **Rebuild**
  - **Resume_rst**

  Restart

Uniform Checkpointer Interface

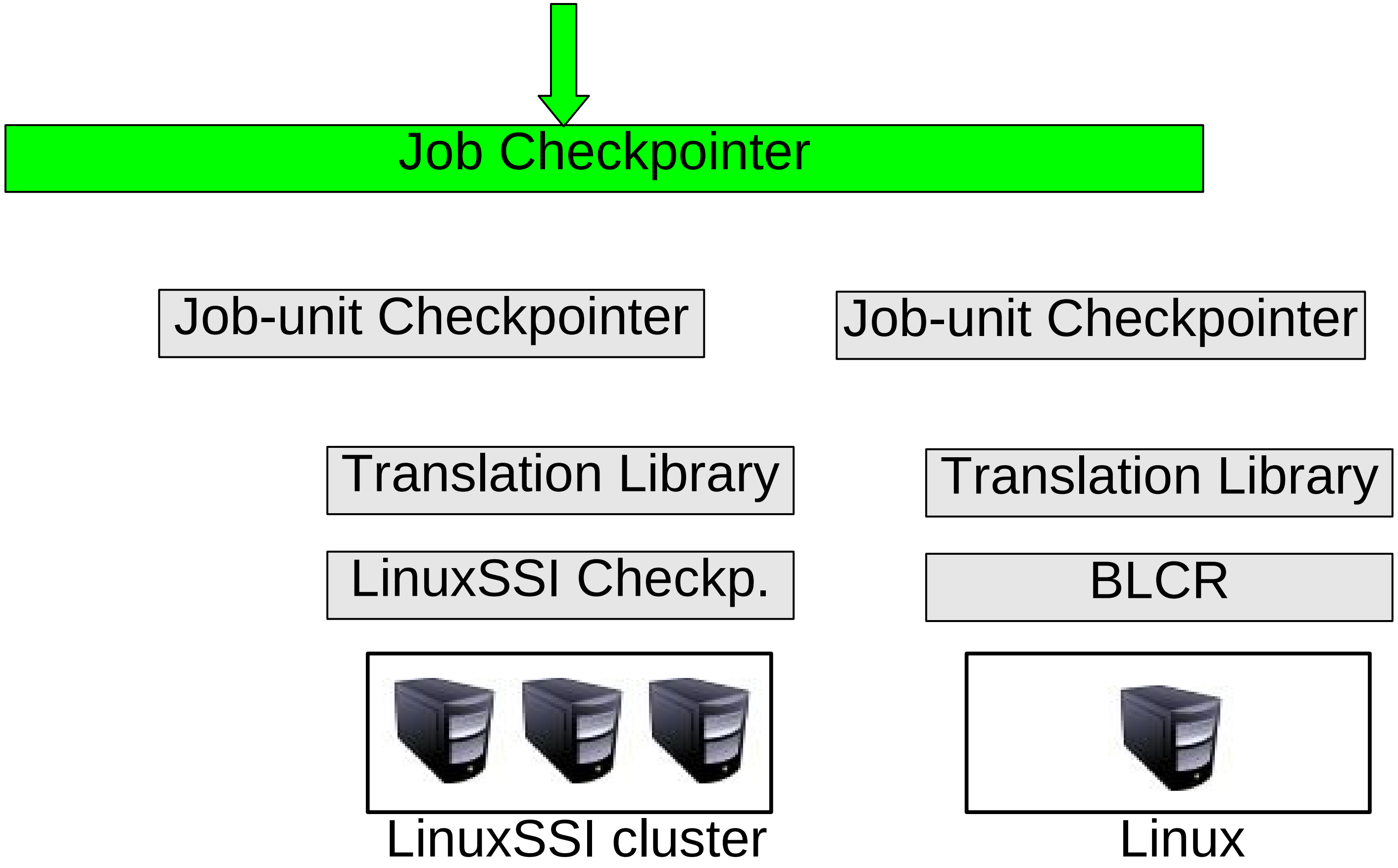
- **Currently, supported checkpointer packages**

- **BLCR**
- **OpenVZ**
- **MTCP**
- **LinuxSSI**
- **(Linux native)**

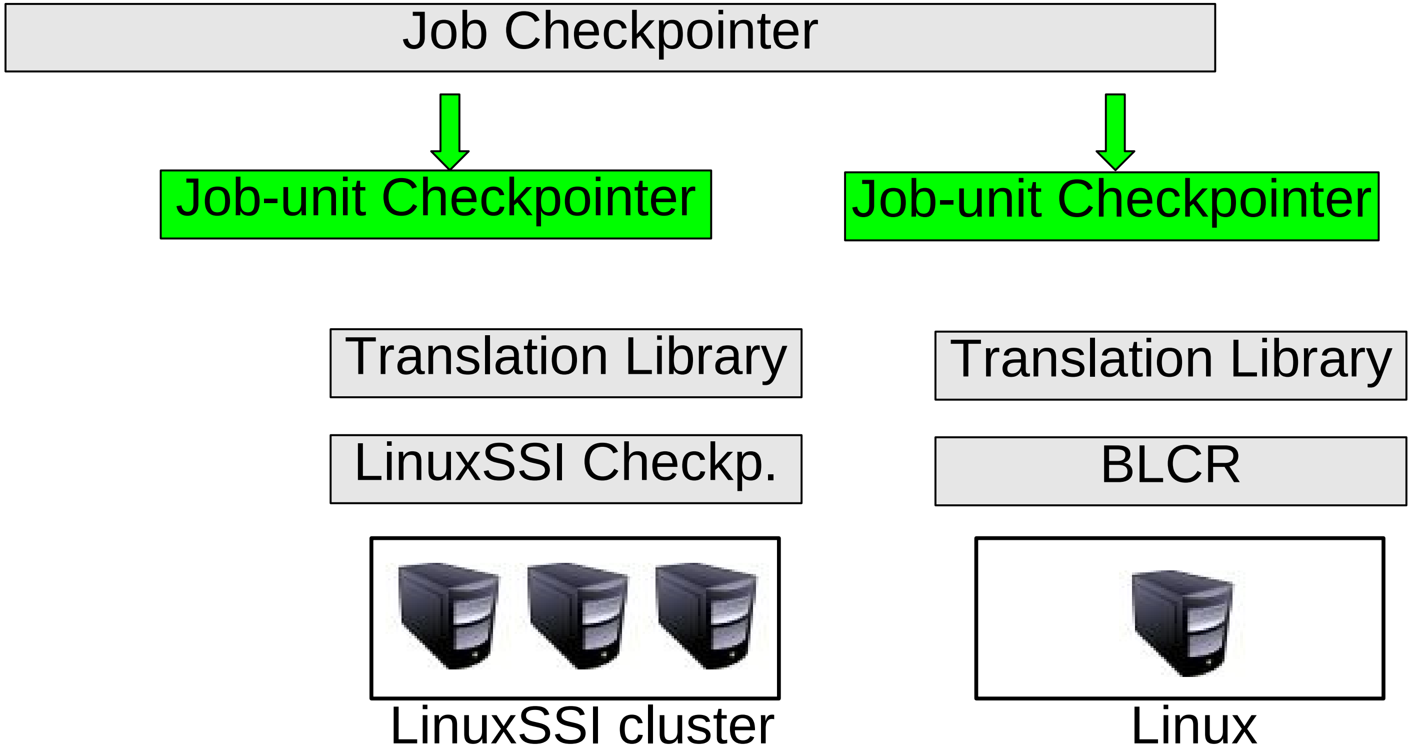
Checkpoint files

- **Must be replicated**
- **And accessible from each grid node**
- **Stored in XtreemFS, providing:**
  - **Stripping**
  - **Automatic replication**
  - **Location-transparent access**
  - **Access control via XtreemOS user accounts**

Coordinated Checkpointing Workflow



Coordinated Checkpointing Workflow



Job Checkpointer

Job-unit Checkpointer

Job-unit Checkpointer

Translation Library

LinuxSSI Checkp.

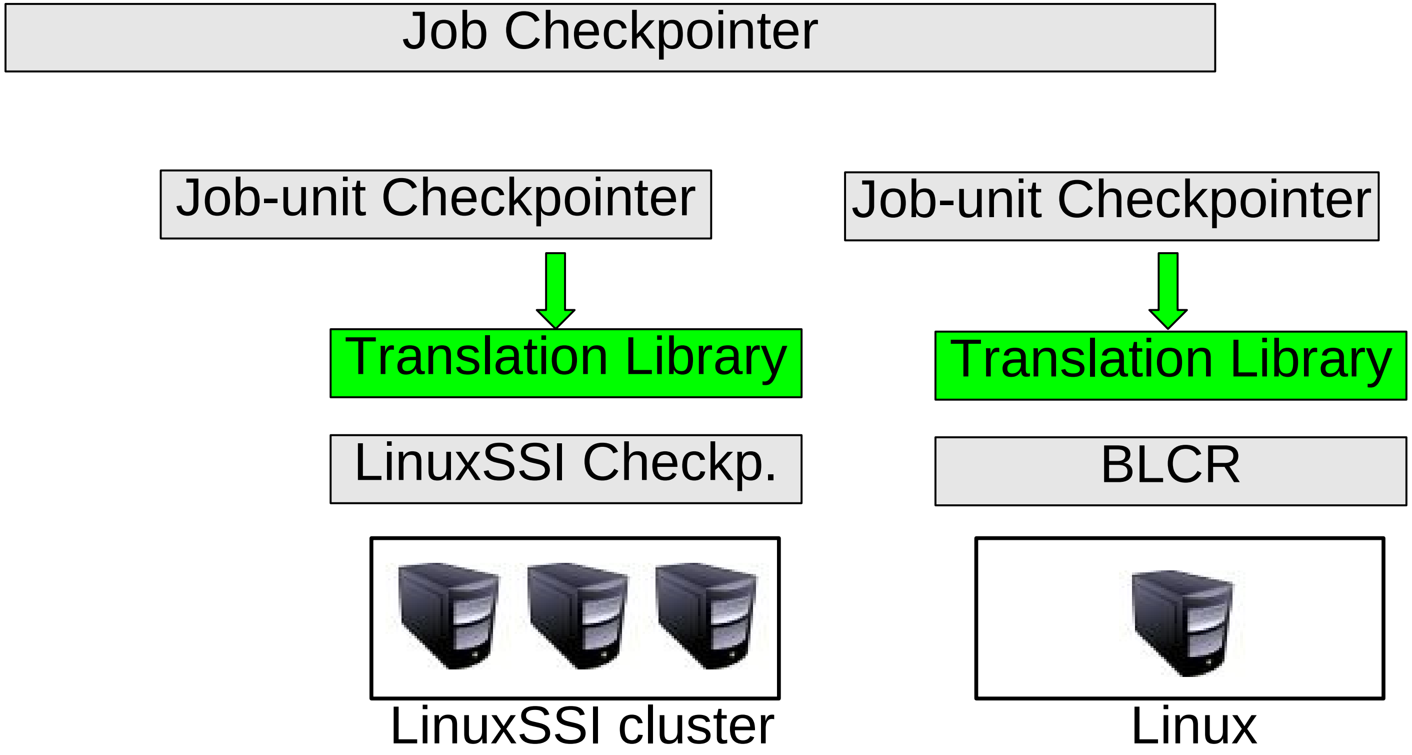
LinuxSSI cluster

Translation Library

BLCR

Linux

Coordinated Checkpointing Workflow



Job Checkpointer

Job-unit Checkpointer

Translation Library

LinuxSSI Checkp.

LinuxSSI cluster

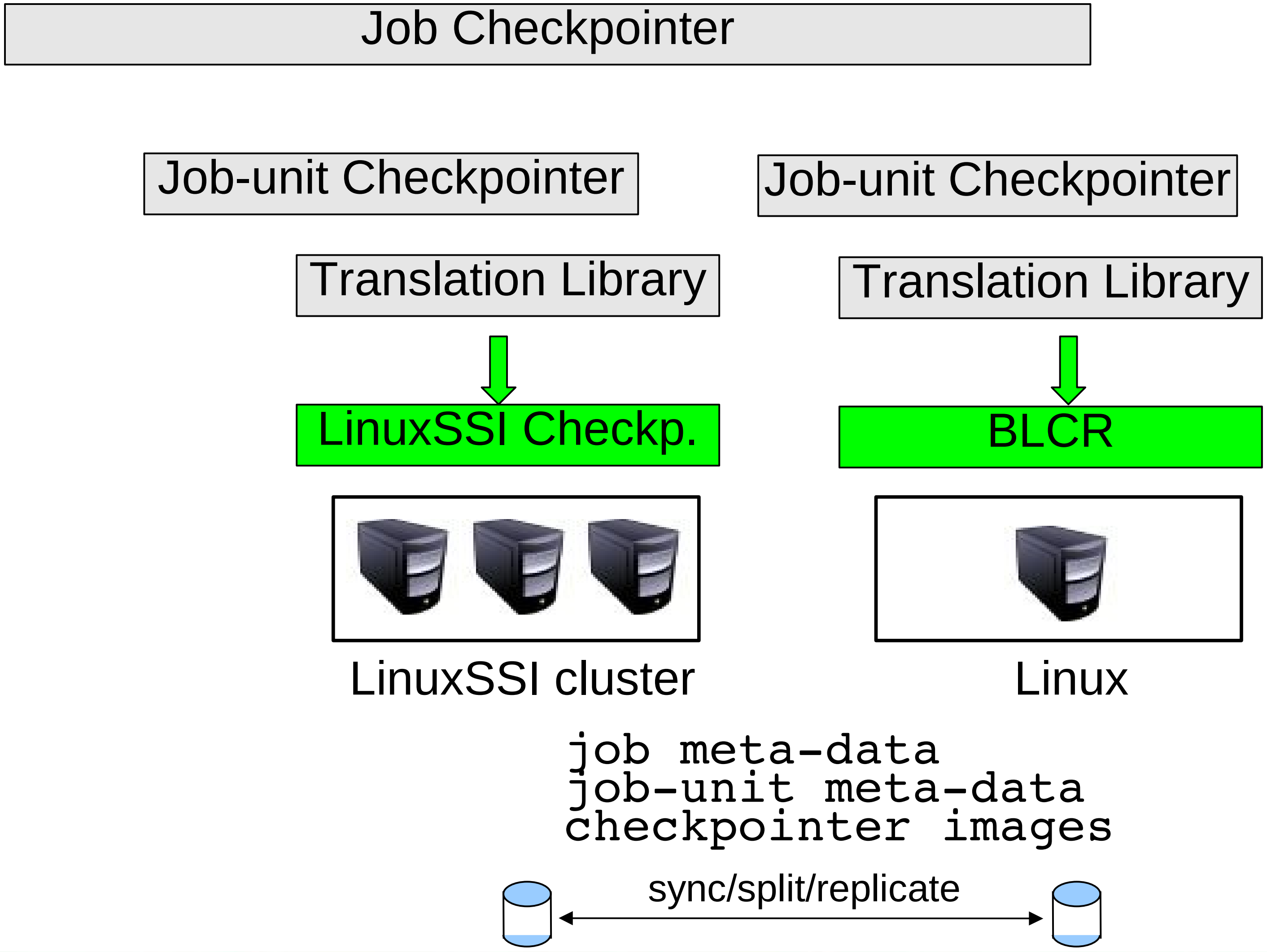
Job-unit Checkpointer

Translation Library

BLCR

Linux

Coordinated Checkpointing Workflow



Job Checkpointer

Job-unit Checkpointer

Job-unit Checkpointer

Translation Library

Translation Library

LinuxSSI Checkp.

BLCR

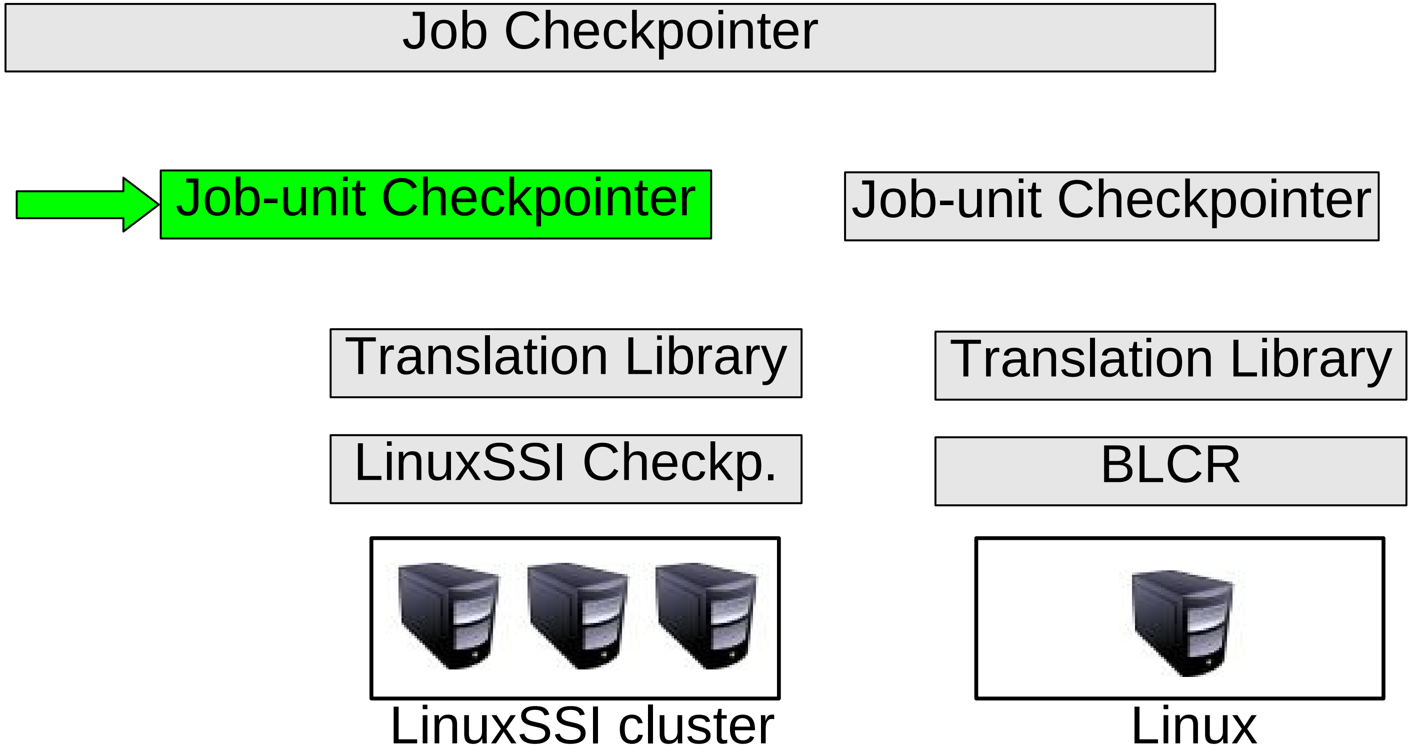
LinuxSSI cluster

Linux

```
job meta-data
job-unit meta-data
checkpointer images
```

sync/split/replicate

Independent Checkpointing Workflow



Job Checkpointer

Job-unit Checkpointer

Job-unit Checkpointer

Translation Library

Translation Library

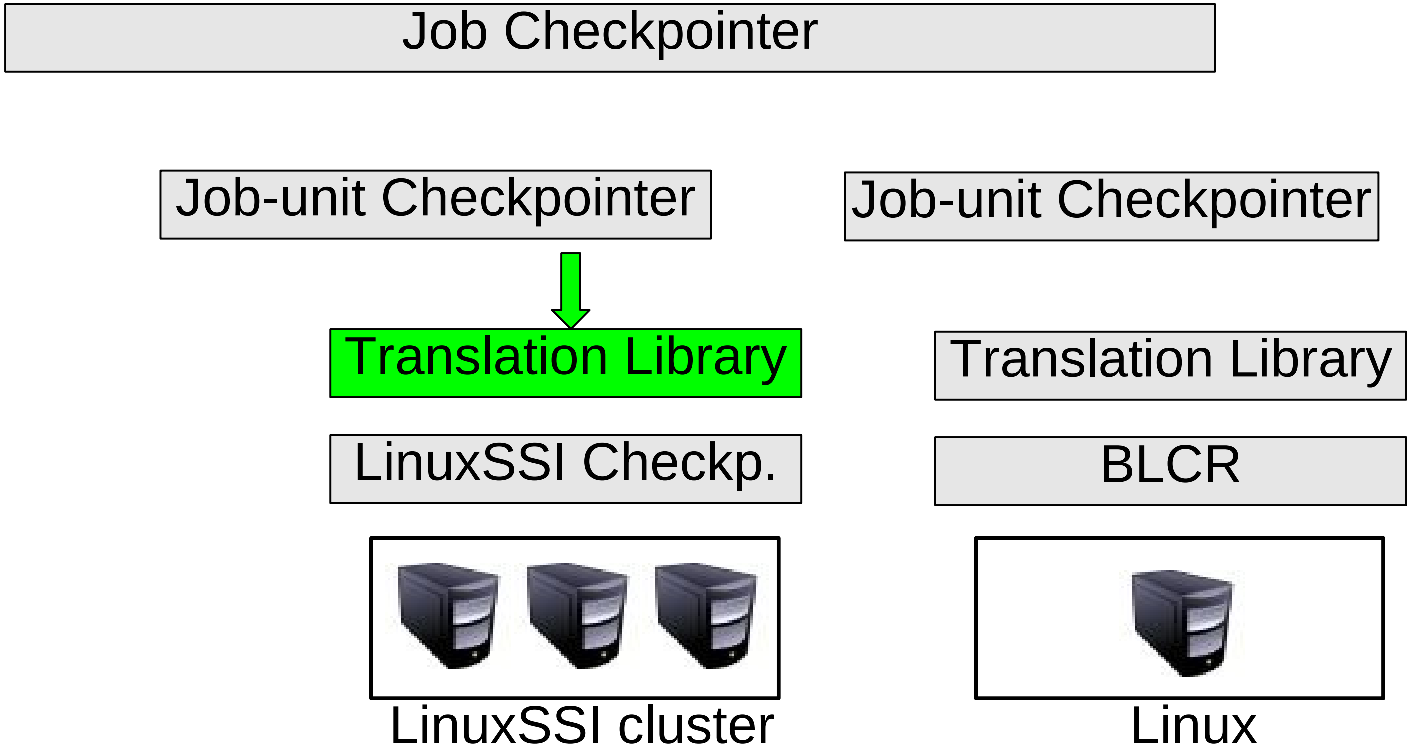
LinuxSSI Checkp.

BLCR

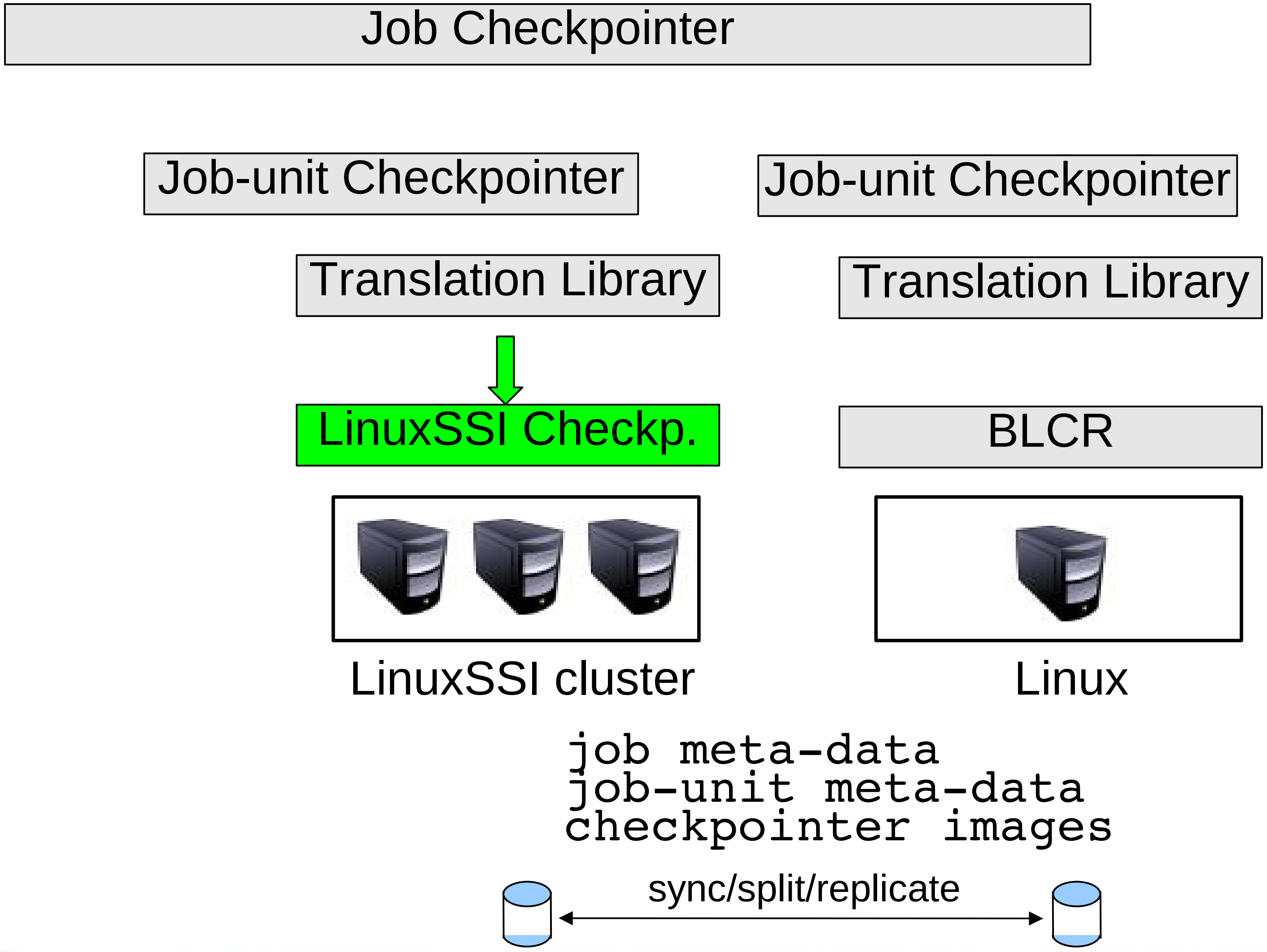
LinuxSSI cluster

Linux

Independent Checkpointing Workflow



Independent Checkpointing Workflow



Job Checkpointer

Job-unit Checkpointer

Job-unit Checkpointer

Translation Library

Translation Library

LinuxSSI Checkp.

BLCR

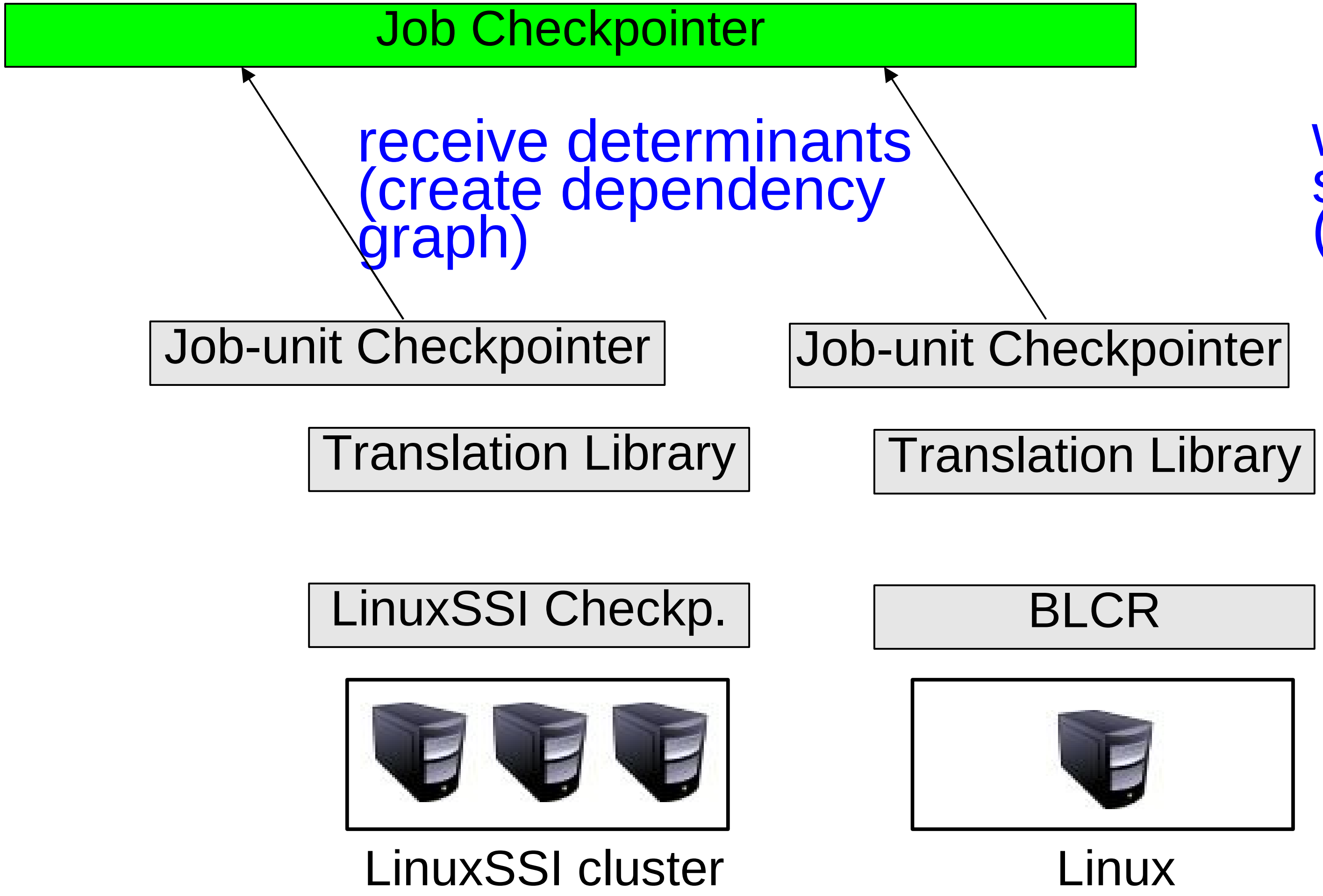
LinuxSSI cluster

Linux

```
job meta-data
job-unit meta-data
checkpointer images
```

sync/split/replicate

Independent Restart Workflow (during application runtime)



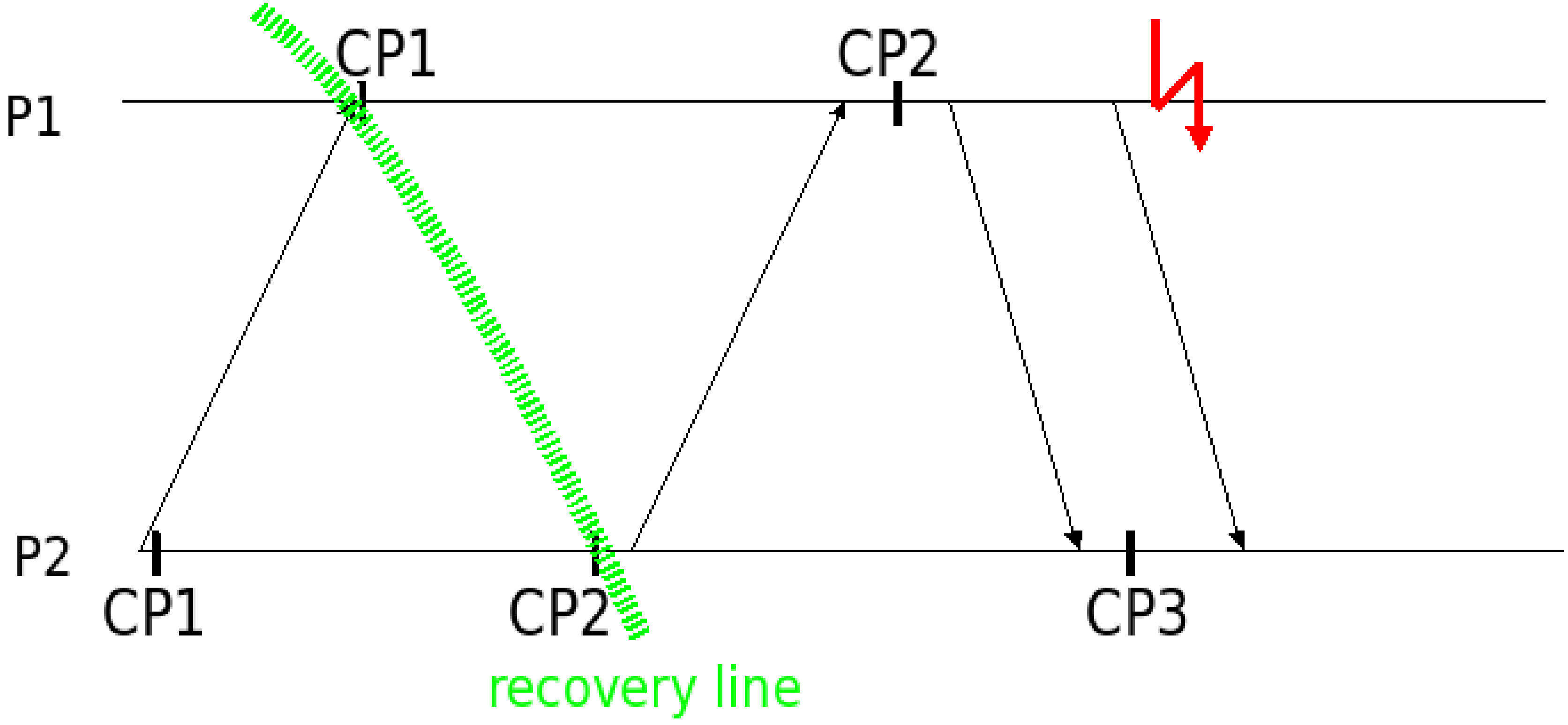
receive determinants (create dependency graph)

wrappers for send, recv, etc. (LD_PRELOAD)

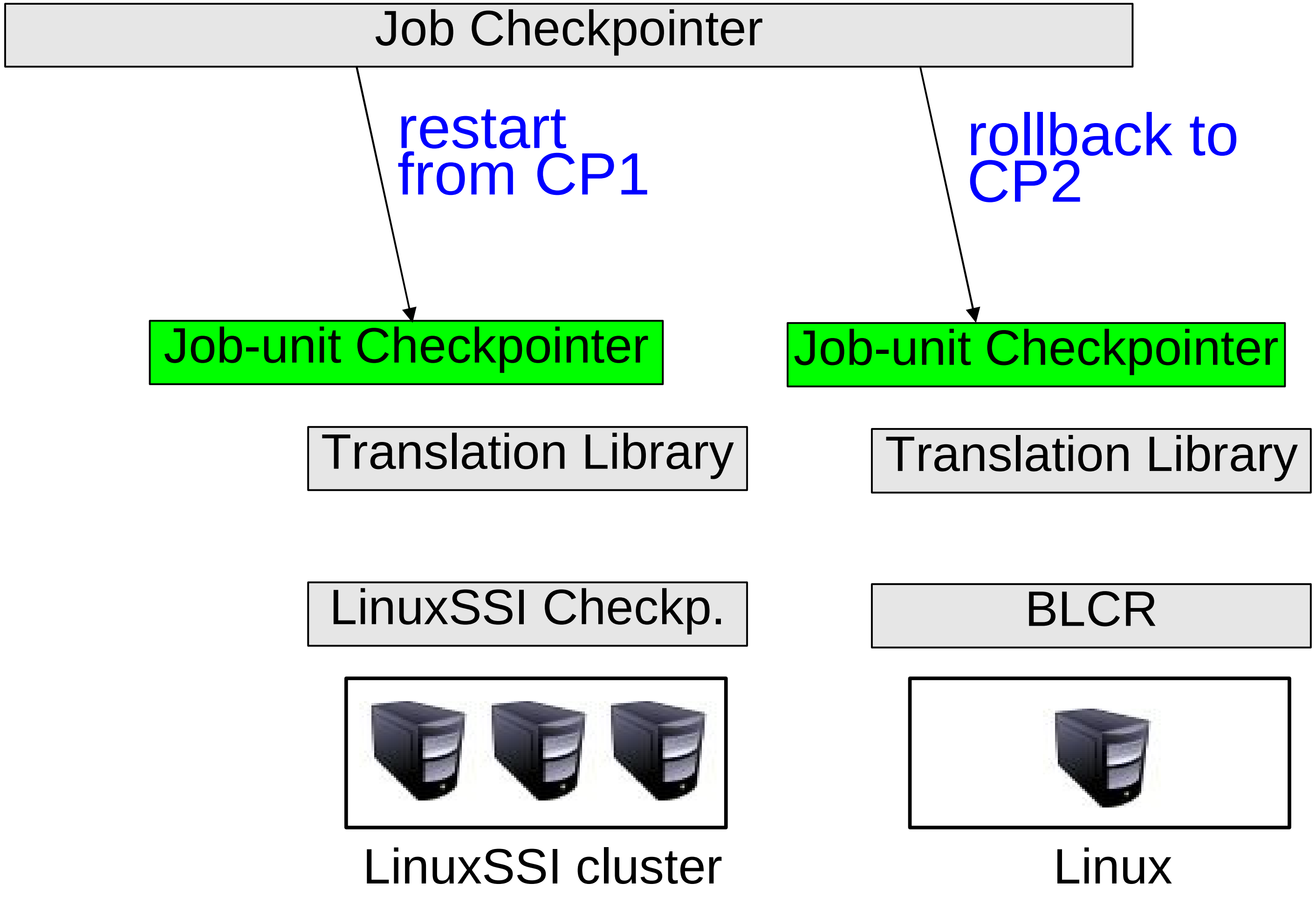
Independent Restart Workflow

Job Checkpointer

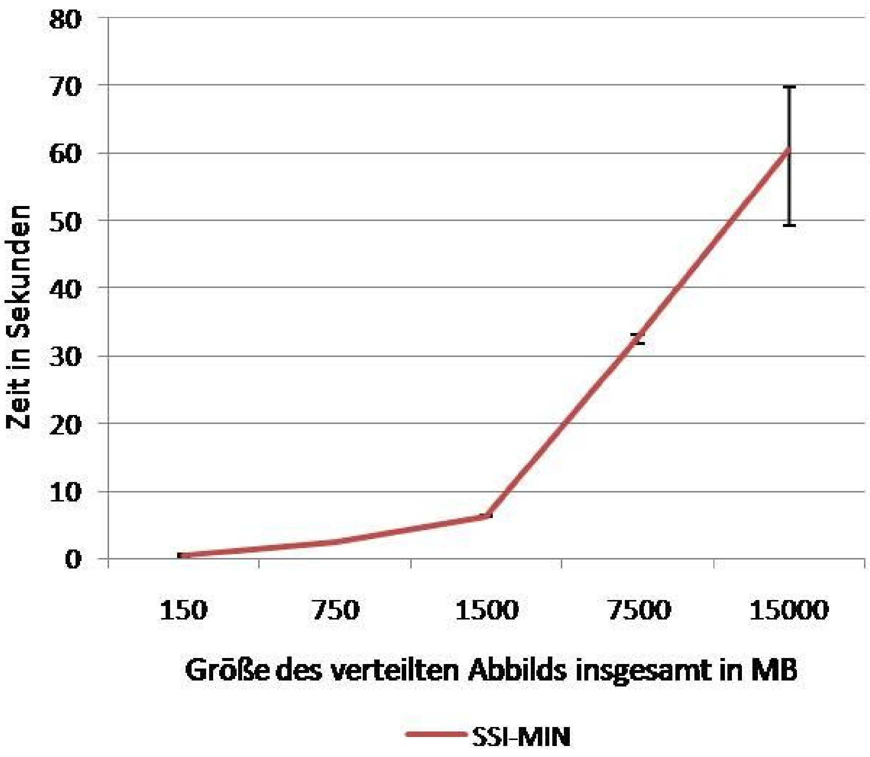
calculate recovery line from received determinants



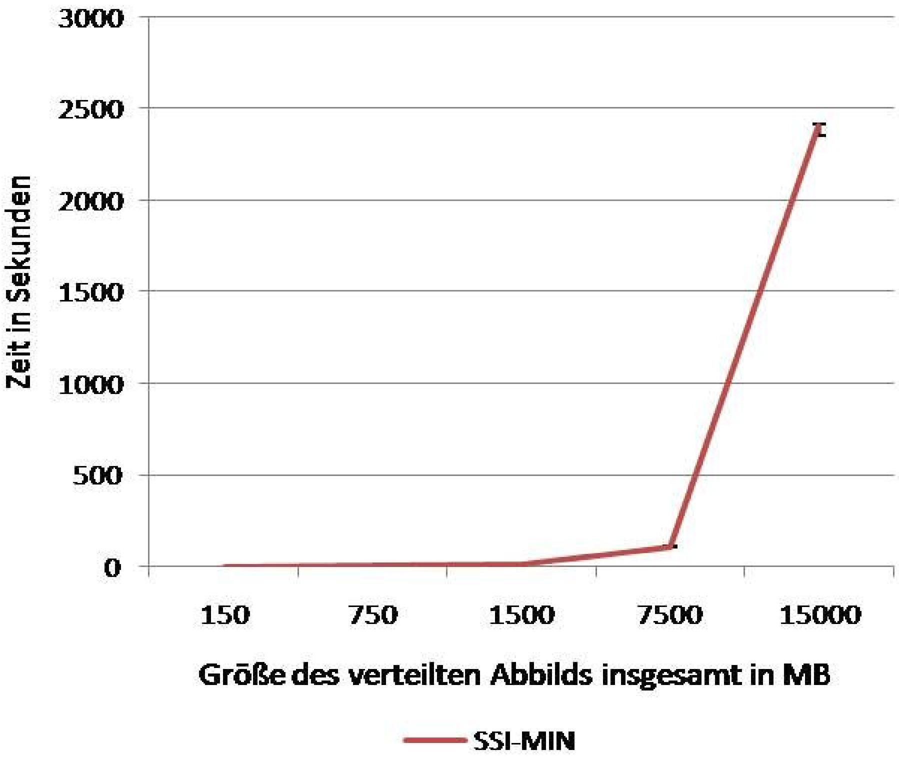
Independent Restart Workflow



Measurements



Checkpoint



Restart

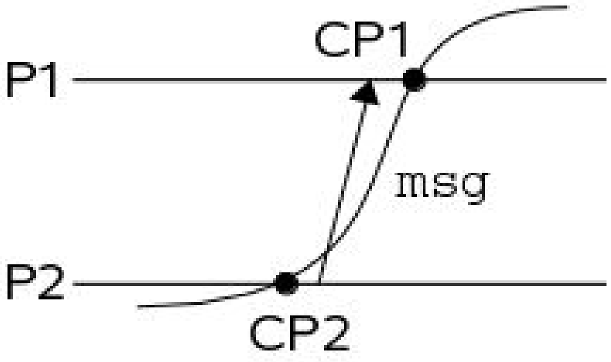
Callback Management

- Implemented in generic part of translib
- Called before and after a checkpoint and after restart
- Common API for application callback registration
- Useful for:
  - Application-level checkpointing
  - Application-level enhancements/optimizations
  - System-level checkpointing of communication channels

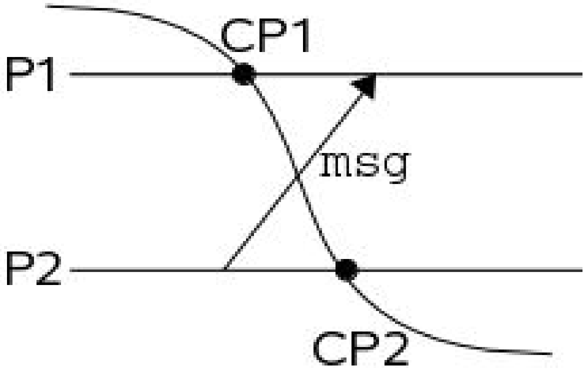
Channel checkpointing with heterogeneous checkpointers

Consistent Checkpoints
- in-transit messages -

- **orphan message**

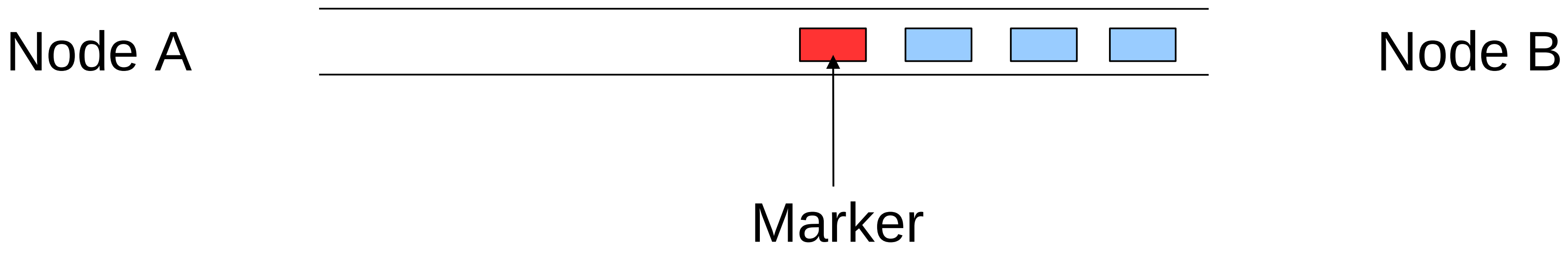


- **lost message:**



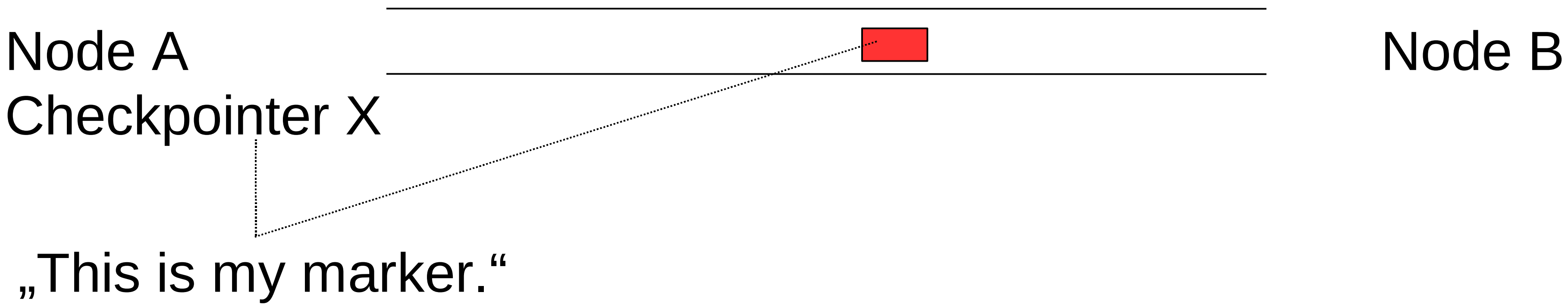
Challenges in the grid context

- **Soluition save in-transit messages**
- **Marker-based approach**



Challenges in the grid context

- **Marker-based approach**
- **Challenges**
  - **incompatible checkpointers must cooperate**
  - **migration support**
  - **transparency (application, checkpointer, operating system)**



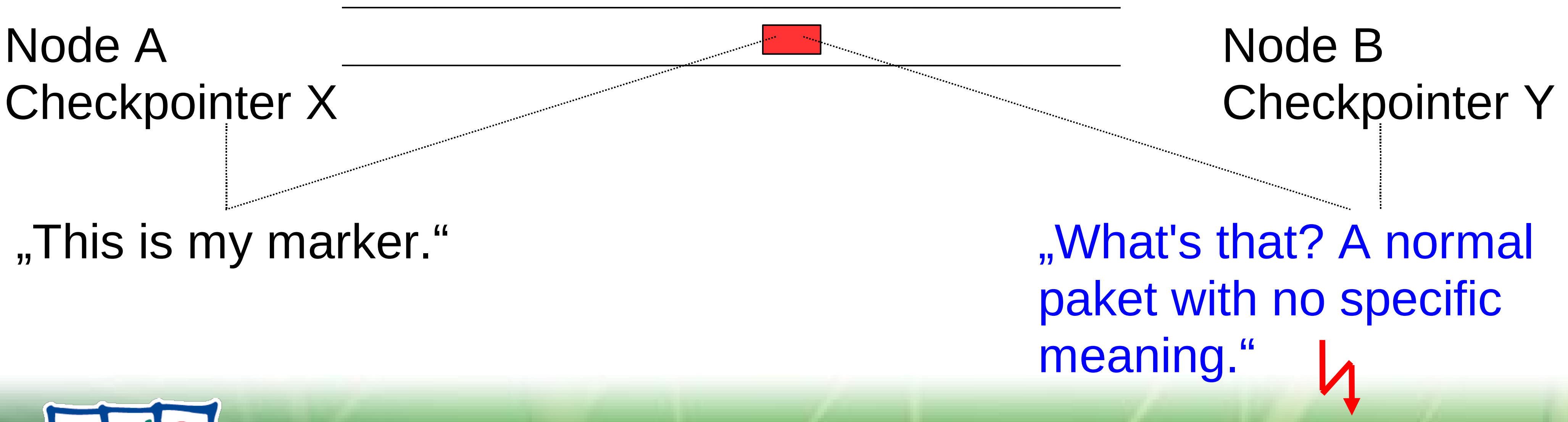
Node A
Checkpointer X

Node B

„This is my marker.“

Challenges in the grid context

- **Marker-based approach**
- **Challenges**
  - **incompatible checkpointers must cooperate**
  - **migration support**
  - **transparency (application, checkpointer, operating system)**



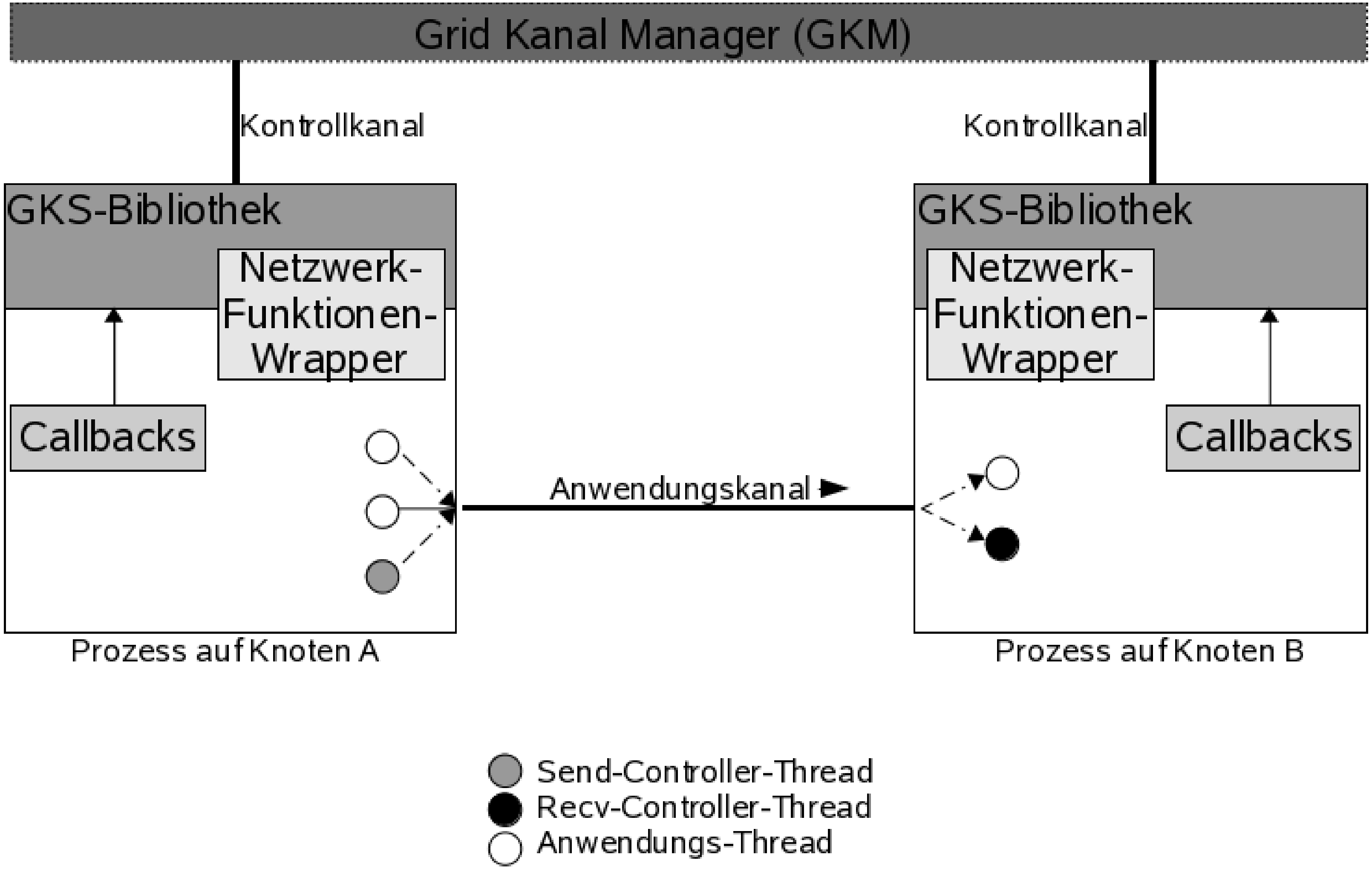
Node A
Checkpointer X

Node B
Checkpointer Y

„This is my marker.“

„What's that? A normal paket with no specific meaning.“ ↯

Architecture



Grid Kanal Manager (GKM)

Kontrollkanal

Kontrollkanal

GKS-Bibliothek

Netzwerk-Funktionen-Wrapper

Callbacks

Anwendungskanal

GKS-Bibliothek

Netzwerk-Funktionen-Wrapper

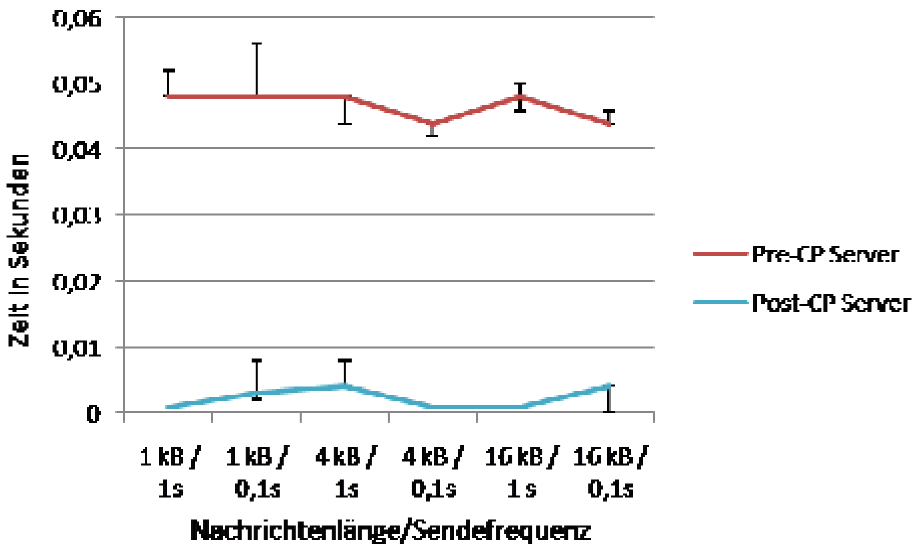
Callbacks

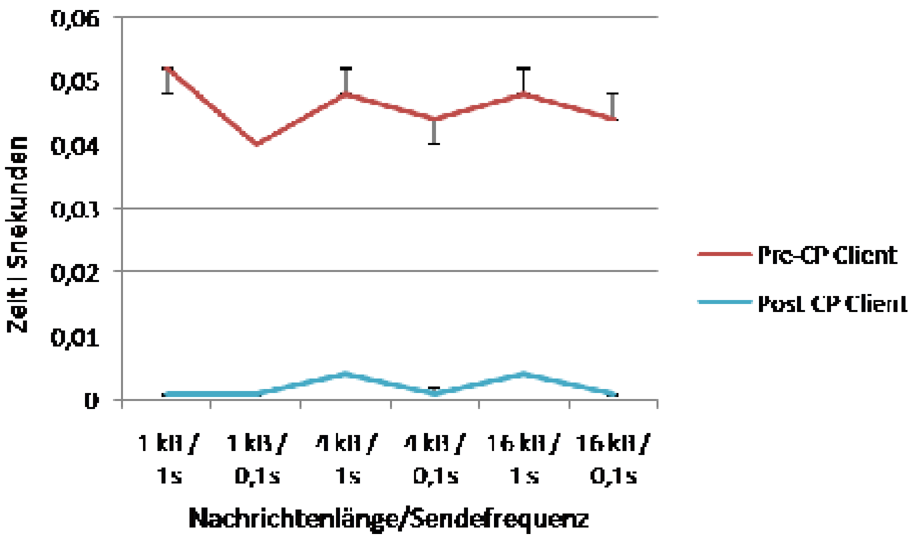
Prozess auf Knoten A

Prozess auf Knoten B

● Send-Controller-Thread
● Recv-Controller-Thread
○ Anwendungs-Thread

Gridkanalsicherung - Messungen -





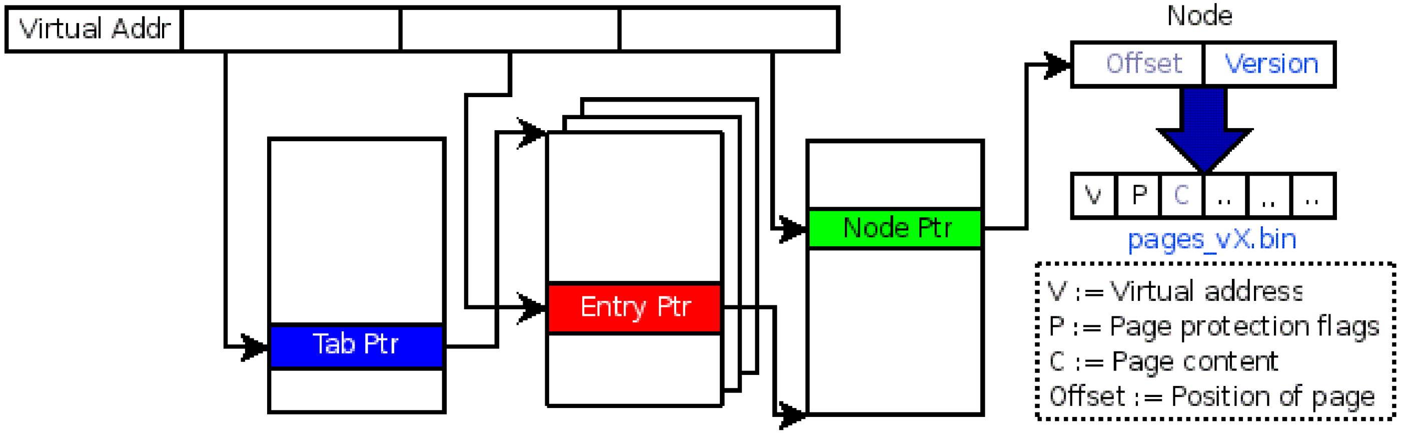
- Nachrichtenlänge und Sendefrequenz ohne Auswirkungen

Adaptive checkpointing

Adaptive Checkpointing
- Incremental Checkpointing -

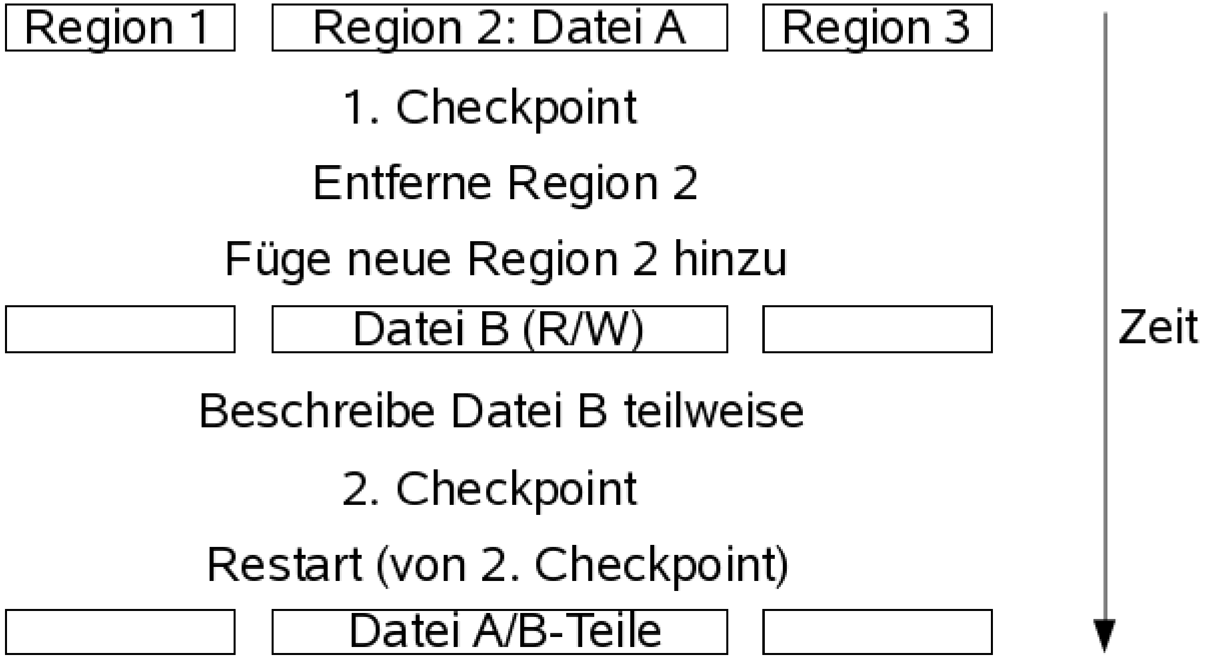
Incremental Checkpointing

- Write-bit
- reflect dynamical memory layout changes
- *mprotect* und *jsdl*



V := Virtual address
P := Page protection flags
C := Page content
Offset := Position of page

Adaptive Checkpointing
- Incremental Checkpointing -



Region 1 | Region 2: Datei A | Region 3

1. Checkpoint

Entferne Region 2

Füge neue Region 2 hinzu

Datei B (R/W)

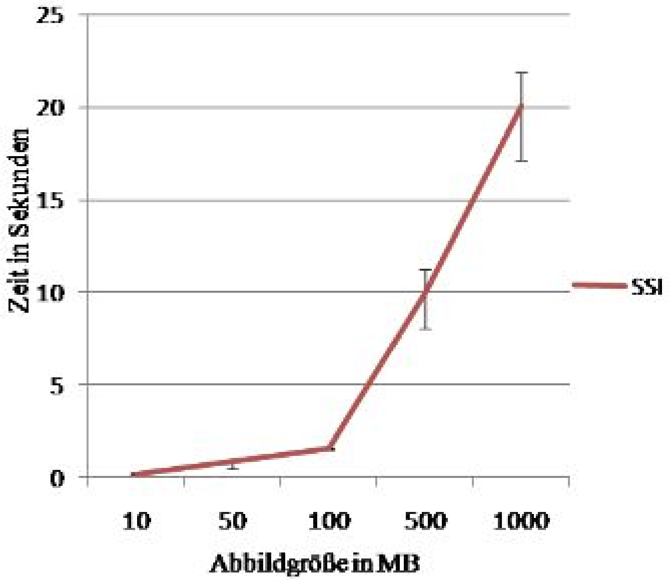
Beschreibe Datei B teilweise

2. Checkpoint

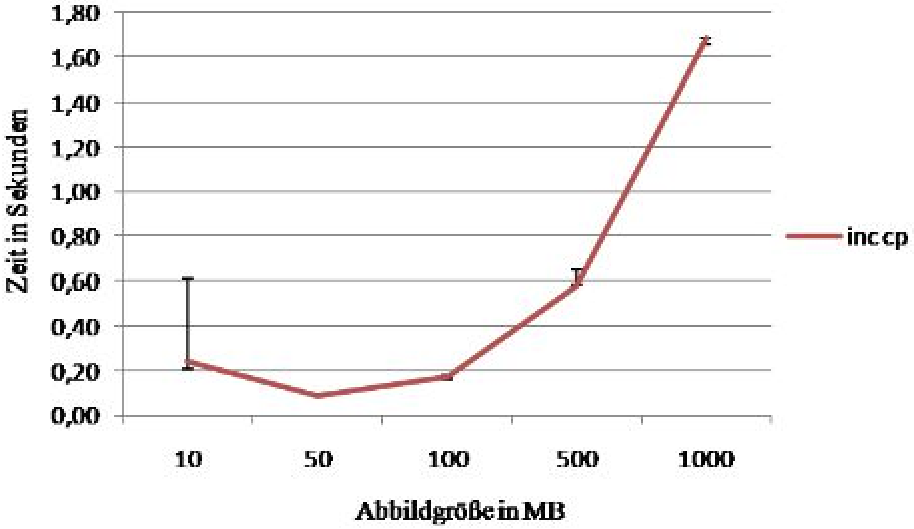
Restart (von 2. Checkpoint)

Datei A/B-Teile

Adaptive Checkpointing
- Incremental Checkpointing -

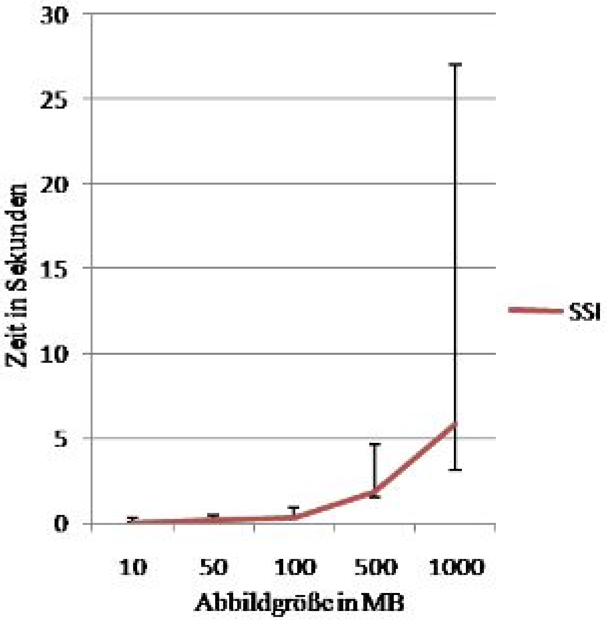


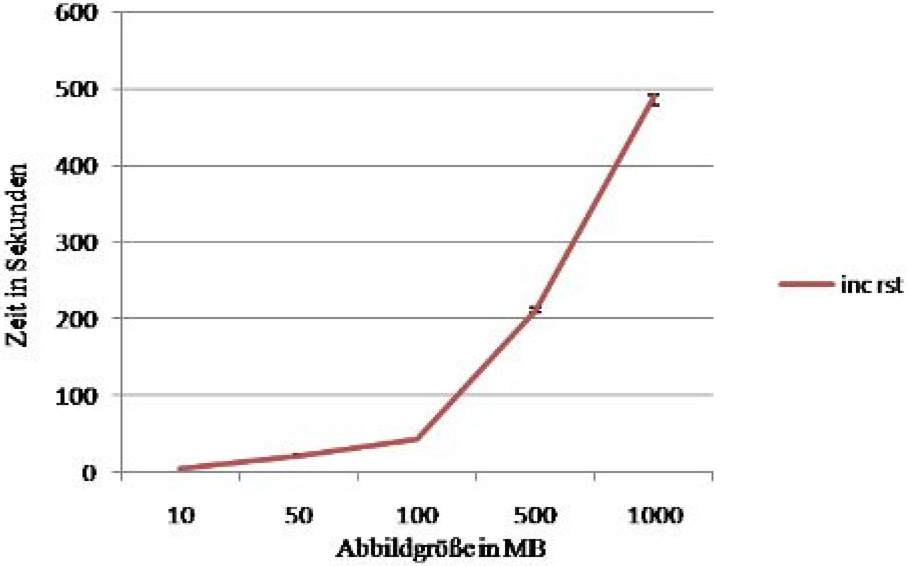
Common Checkpoint



Incremental Checkpoint

Adaptive Checkpointing
- Incremental Checkpointing -





Common Restart

Incremental Restart

Summary

- XtreemGCP offers migration and fault tolerance in grids by providing checkpointing and restart

- It is designed for heterogeneous setups integrating existing checkpointer packages

- Future work:

  virtual machine support & adaptive checkpointing

Acknowledgment

- EC for funding XtreemOS

- XtreemOS- GCP contributors:
  - Heinrich-Heine Universität Düsseldorf
    John Mehnert-Spahn, Eugen Feller
  - INRIA, Rennes, France
    Christine Morin, Thomas Ropars, Surbi Chitre, Stefania Costache