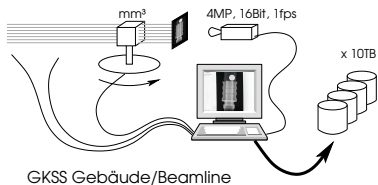


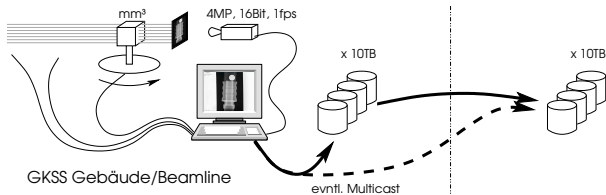
Datenhaltung für die GKSS (Imaging) Beamlines

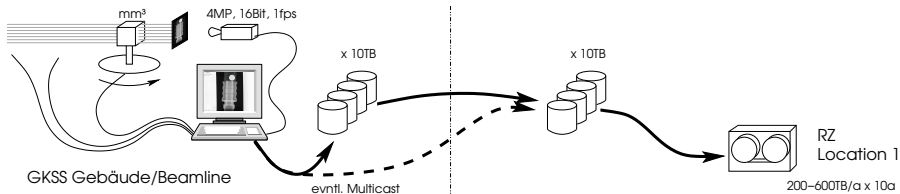
Sven Utcke, GKSS/WPN

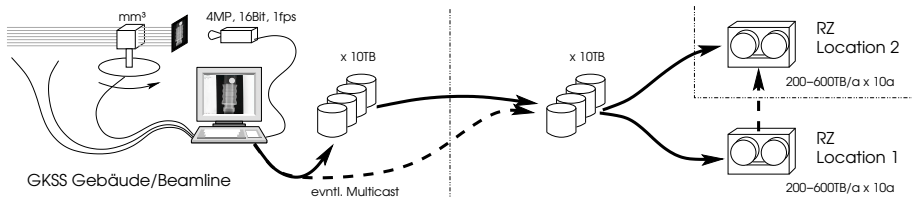
Dienstags-Meeting, 17. Juni 2008

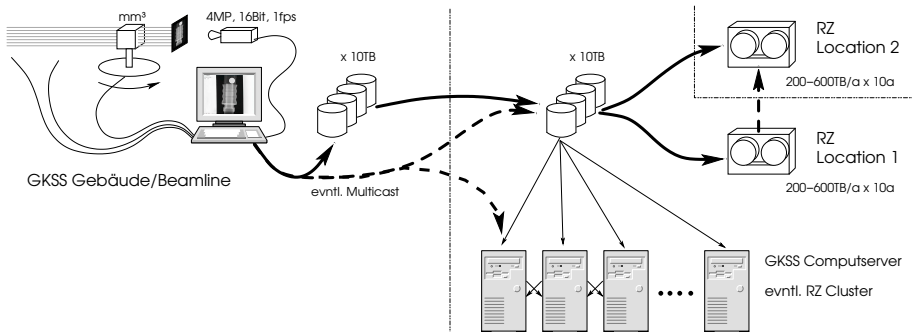


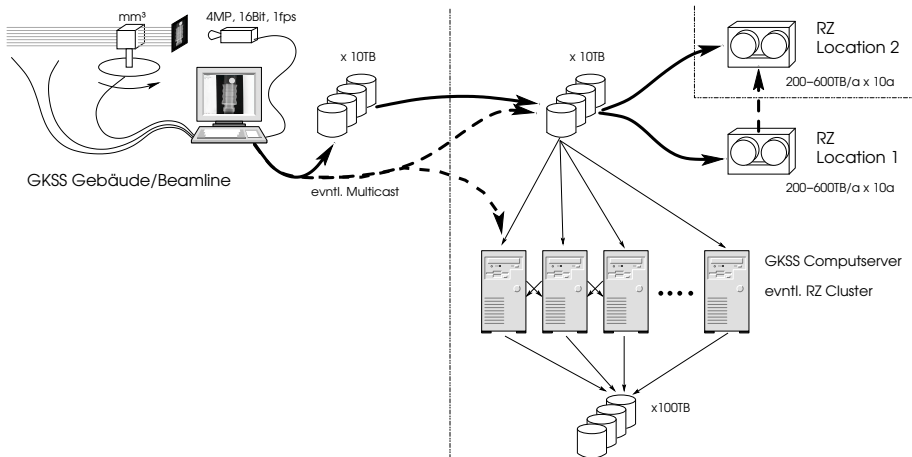


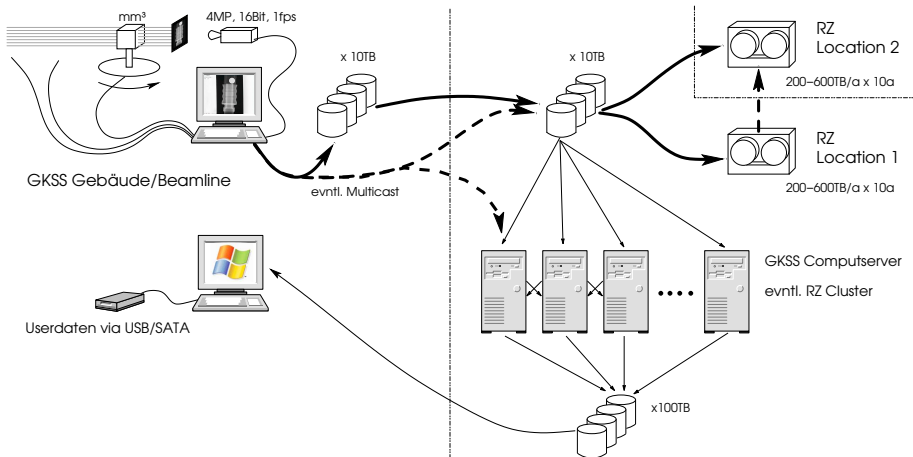


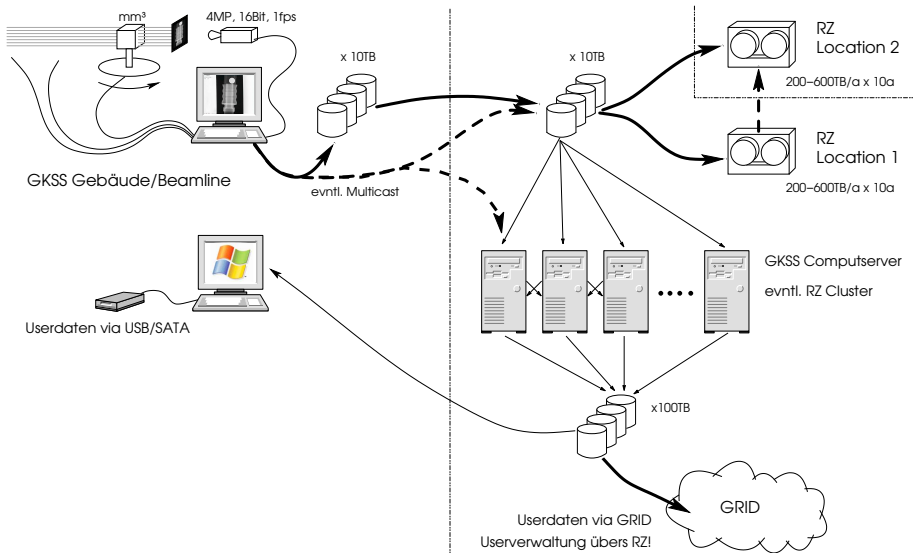


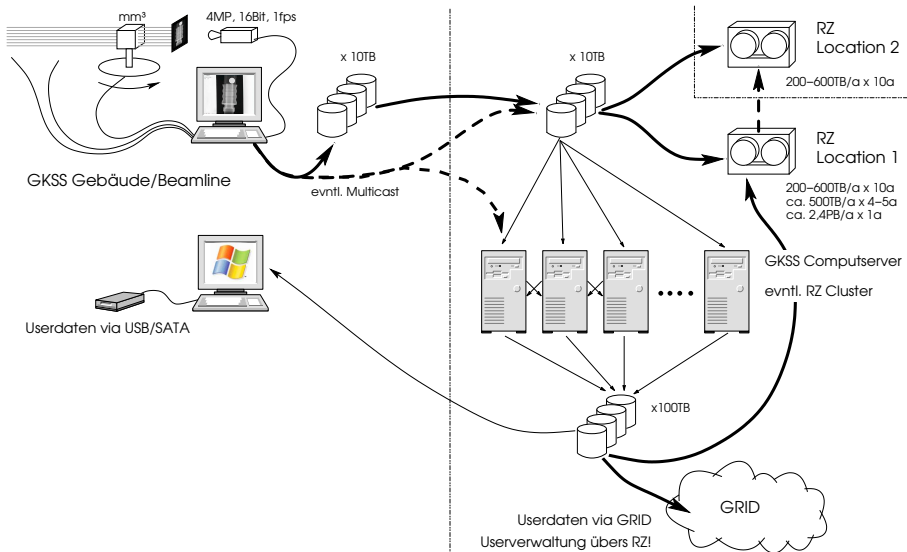


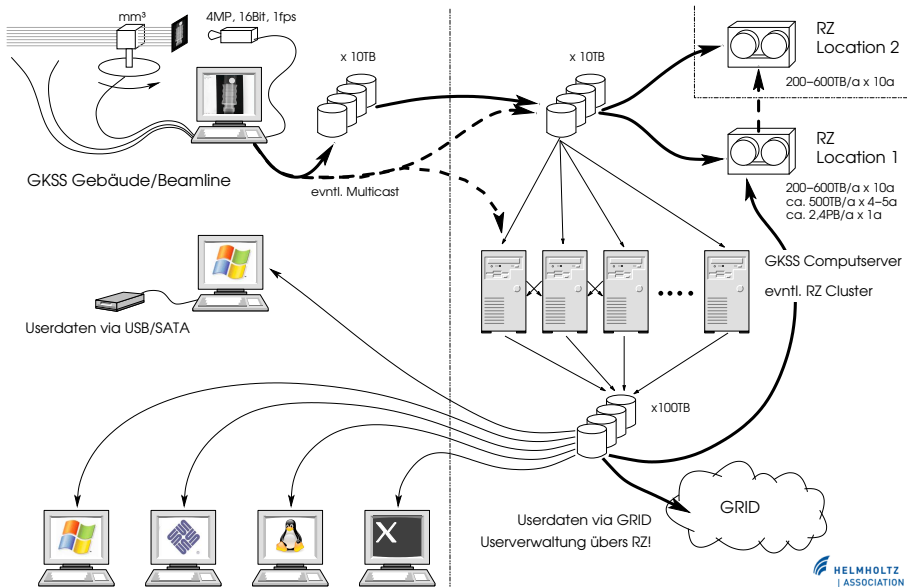


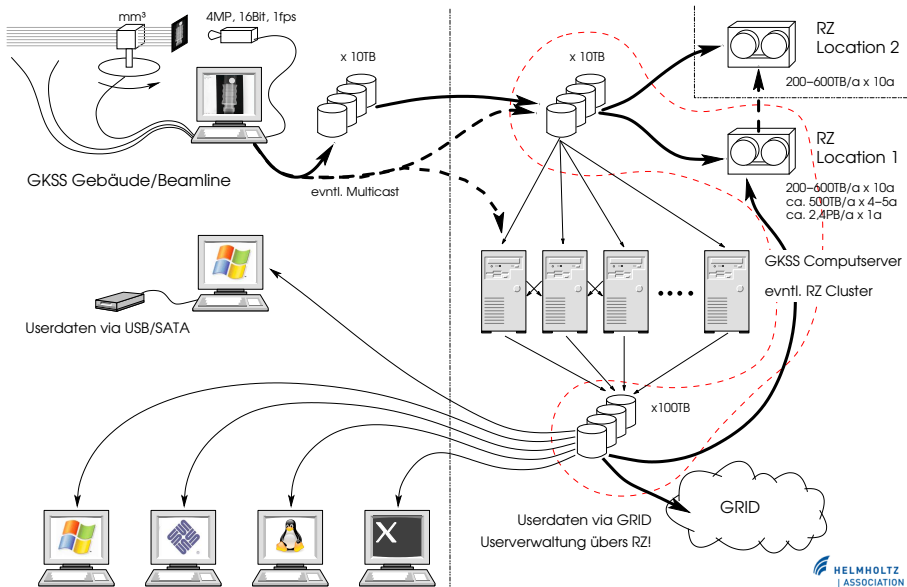












Problem: X PB auf Tape, aber nur $Y \times 10$ TB auf Platte

- ▶ dCache
 - + Vorhanden
 - + Lesender Zugriff von UNIX Systemen aus transparent möglich
 - Zugriff von Windows aus unklar / kompliziert
 - Keine kleinen Dateien (z. Zt. 200 MB), keine Dateien > 2 GB
- ▶ tar/dar im dCache/osm
 - + Vorhanden
 - Zugriff erfordert Kopie in eigenen Cache
 - Externes Bookkeeping erforderlich
 - ▶ welche Dateien in welchem Archiv?
 - ▶ welche Dateien können gelöscht werden?
 - ▶ Zu umständlich!
- ▶ HSM — Hierarchical Storage Management
 - + Für den User komplett transparent
 - Kostet Geld

Problem: X PB auf Tape, aber nur $Y \times 10$ TB auf Platte

▶ dCache

- + Vorhanden
- + **Lesender** Zugriff von UNIX Systemen aus transparent möglich
- Zugriff von Windows aus unklar / kompliziert
- Keine kleinen Dateien (z. Zt. 200 MB), keine Dateien > 2 GB

▶ tar/dar im dCache/osm

- + Vorhanden
- Zugriff erfordert Kopie in eigenen Cache
- Externes Bookkeeping erforderlich
 - ▶ welche Dateien in welchem Archiv?
 - ▶ welche Dateien können gelöscht werden?
- ▶ Zu umständlich!

▶ HSM — Hierarchical Storage Management

- + Für den User komplett transparent
- Kostet Geld

Problem: X PB auf Tape, aber nur $Y \times 10$ TB auf Platte

▶ dCache

- + Vorhanden
- + **Lesender** Zugriff von UNIX Systemen aus transparent möglich
- Zugriff von Windows aus unklar / kompliziert
- Keine kleinen Dateien (z. Zt. 200 MB), keine Dateien > 2 GB

▶ tar/dar im dCache/osm

- + Vorhanden
- Zugriff erfordert Kopie in eigenen Cache
- Externes Bookkeeping erforderlich
 - ▶ welche Dateien in welchem Archiv?
 - ▶ welche Dateien können gelöscht werden?
- ▶ Zu umständlich!

▶ HSM — Hierarchical Storage Management

- + Für den User komplett transparent
- Kostet Geld

Problem: X PB auf Tape, aber nur $Y \times 10$ TB auf Platte

- ▶ dCache
 - + Vorhanden
 - + **Lesender** Zugriff von UNIX Systemen aus transparent möglich
 - Zugriff von Windows aus unklar / kompliziert
 - Keine kleinen Dateien (z. Zt. 200 MB), keine Dateien > 2 GB
- ▶ tar/dar im dCache/osm
 - + Vorhanden
 - Zugriff erfordert Kopie in eigenen Cache
 - Externes Bookkeeping erforderlich
 - ▶ welche Dateien in welchem Archiv?
 - ▶ welche Dateien können gelöscht werden?
 - ▶ Zu umständlich!
- ▶ HSM — Hierarchical Storage Management
 - + Für den User komplett transparent
 - Kostet Geld

Problem: X PB auf Tape, aber nur $Y \times 10$ TB auf Platte

- ▶ dCache
 - + Vorhanden
 - + **Lesender** Zugriff von UNIX Systemen aus transparent möglich
 - Zugriff von Windows aus unklar / kompliziert
 - Keine kleinen Dateien (z. Zt. 200 MB), keine Dateien > 2 GB
- ▶ tar/dar im dCache/osm
 - + Vorhanden
 - Zugriff erfordert Kopie in eigenen Cache
 - Externes Bookkeeping erforderlich
 - ▶ welche Dateien in welchem Archiv?
 - ▶ welche Dateien können gelöscht werden?
 - ▶ Zu umständlich!
- ▶ HSM — Hierarchical Storage Management
 - + Für den User komplett transparent
 - Kostet Geld ... oder nicht?

- ▶ **Kaufen**
 - + Einfach, schnell, zuverlässig
 - + Funktioniert von jedem OS aus
 - Kostet Geld
 - Kein Know How im Haus
- ▶ **Frickeln** — ein Frontend für dCache/osm
 - + 100 % auf unsere Bedürfnisse zugeschnitten
 - + Funktioniert von jedem OS aus
 - + Proof of Konzept existiert
 - ? Know How im Haus — aber eben nur hier
 - Zuverlässigkeit unerprobt
 - Kostet Arbeitskraft
- ▶ **Aufrüsten** — den dCache direkt nutzbar machen
 - + Baut auf vorhandener Infrastruktur auf
 - + Know How im Haus
 - ? Transparenter Zugriff nur unter UNIX (und dort umständlich)
 - Kostet Arbeitskraft — RZ hat keine Kapazitäten?

- ▶ **Kaufen**
 - + Einfach, schnell, zuverlässig
 - + Funktioniert von jedem OS aus
 - Kostet Geld
 - Kein Know How im Haus
- ▶ **Frickeln — ein Frontend für dCache/osm**
 - + 100% auf unsere Bedürfnisse zugeschnitten
 - + Funktioniert von jedem OS aus
 - + Proof of Konzept existiert
 - ? Know How im Haus — aber eben nur hier
 - Zuverlässigkeit unerprobt
 - Kostet Arbeitskraft
- ▶ **Aufrüsten — den dCache direkt nutzbar machen**
 - + Baut auf vorhandener Infrastruktur auf
 - + Know How im Haus
 - ? Transparenter Zugriff nur unter UNIX (und dort umständlich)
 - Kostet Arbeitskraft — RZ hat keine Kapazitäten?

- ▶ **Kaufen**
 - + Einfach, schnell, zuverlässig
 - + Funktioniert von jedem OS aus
 - Kostet Geld
 - Kein Know How im Haus
- ▶ **Frickeln — ein Frontend für dCache/osm**
 - + 100 % auf unsere Bedürfnisse zugeschnitten
 - + Funktioniert von jedem OS aus
 - + Proof of Konzept existiert
 - ? Know How im Haus — aber eben nur hier
 - Zuverlässigkeit unerprobt
 - Kostet Arbeitskraft
- ▶ **Aufrüsten — den dCache direkt nutzbar machen**
 - + Baut auf vorhandener Infrastruktur auf
 - + Know How im Haus
 - ? Transparenter Zugriff nur unter UNIX (und dort umständlich)
 - Kostet Arbeitskraft — RZ hat keine Kapazitäten?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ wem gehören die Daten?
- ▶ id's werden eventuell neu vergeben ⇒ wem gehören die Daten?

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?

Selbstverpflichtung der Aufbewahrung für 10 Jahre

Problem:

- ▶ Daten „gehören“ einer (u)id
- ▶ Zugang zu den Daten haben (Gruppen von) weiteren id's
- ▶ Die id's sind einer natürlichen Person (Hans Meier) oder einer Funktion (postmaster) zugeordnet
- ▶ Dieses Mapping erfolgt z.Zt. in der Registry
- ▶ Accounts gelten nur wenige Monate ⇒ **wem gehören die Daten?**
- ▶ id's werden eventuell neu vergeben ⇒ **wem gehören die Daten?**

Eventuell kann uns das GRID den Weg weisen?

Wie werden dort User identifiziert?