

Pflichtenheft zum Projekt

Aufbau einer Cloud-Infrastruktur in einem Forschungszentrum



**Hochschule für Technik
und Wirtschaft Berlin**

University of Applied Sciences



bearbeitet von:	Tom Schubert
Studiengang:	Angewandte Informatik (Master)
Fachbereich:	Wirtschaftswissenschaften II
Matrikelnummer:	535279
zuständiger Prof.:	Prof. Dr. Hermann Heßling
zuständige Mitarbeiter (DESY):	Patrick Furhmann Yves Kemp
Datum (Version):	07.05.2015 (Ver. 1.0)

Contents

1	Projektziel	3
2	Anforderungen	4
2.1	Muss-Anforderungen	4
2.2	Optionale Anforderungen	4
2.3	Abgrenzung	4
3	Arbeitspakete	5
3.1	Vorbereitung	5
3.2	Machbarkeitsprüfung	5
3.2.1	Container & Infiniband	5
3.2.2	Erstellen eines Images	6
3.2.3	Shared File System	6
3.3	Sicherheit	6
3.3.1	Root Access	6
3.3.2	Sandboxing	6
3.3.3	Einschränkung der Kommunikation von verschiedenen Netzen . . .	7
3.4	Geschwindigkeit (Performance)	7
3.4.1	Benchmarking	7
3.4.2	Overhead	7
3.5	Zukunftsaussichten	7
3.5.1	User Support	7
3.5.2	generic System	8
3.5.3	Orchestrierung	8
4	Projektplan	9
5	Zeitplanung	10

1 Projektziel

Ziel des Projektes ist es zu **untersuchen**, in wie fern es möglich ist auf einem HPC-Cluster die Container-Technologie zum Einsatz zu bringen.

Forschungsgruppen am DESY (Deutsches Elektronen-Synchrotron) verwendet verschiedene Tools und Applikationen für die Verarbeitung ihrer Daten, welche zumeist die Größe der zu verarbeitenden Daten herkömmlicher Rechner überschreiten. Hierbei sind für die einzelnen Anwendungen verschiedene Voraussetzungen gegeben, wie z. B. bestimmte Betriebssysteme oder Bibliotheken in festgelegten Versionen.

Da die Forschungsgruppen und damit auch die Anwendungen und Anforderungen stetig wechseln ist es nicht möglich den Anwendern eine einzige Umgebung zur Verfügung zu stellen.

Es ist gefordert eine Lösung zu finden, die sowohl auf der einen Seite dem Anwender die Möglichkeit bietet - so leicht wie möglich - seine Anwendung zu starten, auf der anderen Seite dennoch die Performance des HPC-Clusters nicht zu stark beeinträchtigt.

Darüber hinaus ist davon auszugehen, dass Anwender des HPC-Clusters, die dieses für Forschungszwecke nutzen, im Bereich der Informatik **nicht** versiert sind.

Es ist dem entsprechend gefordert, dass die zu erstellende Lösung so Simpel wie möglich zu handhaben ist.

2 Anforderungen

Folgend werden die Anforderungen des Projektes erläutert und diese in verschiedene Prioritätsstufen untergliedert. Dabei sind Muss-Anforderungen jene, die das Projektziel im ganzen Beeinflussen und zwingend für den erfolgreichen Abschluss des Projektes fertiggestellt werden müssen.

2.1 Muss-Anforderungen

Untersuchung der ...

- Anwendungsmöglichkeit von Containern im HPC-Umfeld
- Möglichkeit von Verwendung des Infinibandes mit Containern, welches im HPC-Clusters bei der Kommunikation zum Einsatz kommt
- Performanceeinbußen bei Verwendung von Container-Lösungen (Benchmarking)
- Möglichkeit zur Verteilung dieser Container mit
- Sicherheitsaspekte von Containern (Sandboxing)
- benötigten Berechtigungen zum starten eines Containers

2.2 Optionale Anforderungen

Prüfen ob ...

- die Möglichkeit besteht von extern in Container einzugreifen, um dem Anwender zusätzlichen Support bieten zu können
- es möglich ist, ein generisches System zu schaffen welches automatisch aufgrund gestellter Anforderungen den passenden Container zusammenstellt.

2.3 Abgrenzung

Der Einsatz von virtuellen Maschinen (VMs) ist nicht Ziel des Projektes und ist auch nicht zu untersuchen. Virtuelle Maschinen erzeugen ein zu großes Overhead um in großen Verteilten Systemen - wie dem HPC-Clusters - zum Einsatz zu kommen.

3 Arbeitspakete

Für eine einfachere Übersicht werden im folgenden Abschnitt einzelnen Projektschritte in Pakete zusammengefasst. Innerhalb der Pakete werden die einzelnen Schritte detailliert beschrieben um das Ziel dieses Schrittes festzulegen. Diese Zusammenfassung bedeutet nicht, dass die einzelnen Arbeitsschritte stets in der hier angegebenen Reihenfolge stattfinden.

Ein erster Ablaufplan befindet sich im Kapitel Projektplan.

3.1 Vorbereitung

In diesem Abschnitt werden nötigen Vorbereitungen für das Projekt getroffen. Dazu gehören folgende Unterpunkte:

- Hardware- & Softwarevoraussetzungen schaffen (Berechtigungen beantragen, Fernzugriffe einrichten, ...)
- Literaturrecherche über verschiedene Container-Lösungen
- Literaturrecherche über HPC-Cluster

3.2 Machbarkeitsprüfung

Im Arbeitspaket Machbarkeitsprüfung soll geprüft werden, ob verschiedene Anforderungen zu realisieren sind. Wichtig ist, dass lediglich die Prüfung und nicht die Umsetzung gefordert ist.

Als Ziel einzelner Schritte dieses Abschnittes können Dokumentationen zur Anweisung über die Realisierung erstellt werden (Anweisungsbeschreibung).

3.2.1 Container & Infiniband

Das Infiniband ist die Hauptkommunikation zwischen den Rechnerknoten des HPC-Clusters und stellt somit einen wichtigen Aspekt bei der Untersuchung von Container-Technologien dar.

In diesem Arbeitsabschnitt soll geprüft werden, ob die Möglichkeit besteht Container in einem Umfeld von Infiniband-Kommunikation einzusetzen.

Dieser Projektabschnitt ist Grundvoraussetzung für das ganze Projekt und wird aus diesem Grund an erster Stelle durchgeführt (Bereits in der Vorbereitungsphase des Projektes). Ohne diese Möglichkeit sind keine weiteren Untersuchungen im Bereich von

Containern notwendig. Es ist anschließend gefordert weitere Alternativen zu finden, die mit der Problemstellung umgehen können.

3.2.2 Erstellen eines Images

Es soll eine Anweisung für den Anwender geschrieben werden welche erklärt, wie der Anwender vorgehen muss um ein entsprechendes Image - mit seiner benötigten Software - zu erstellen (Userdokumentation).

3.2.3 Shared File System

Um die Daten, welche ausgewertet werden sollen, zu erreichen existieren verschiedene File-Systems. Es soll untersucht werden wie diese Daten in den Container geladen werden können und welche Performance sich dabei ergibt.

Als Idee sind folgende zwei Varianten möglich:

- Der Container unterstützt alle nötigen File-Systeme und kann auf die Daten direkt zugreifen (Nachteil: Alle Module zum Lesen und Verarbeiten verschiedener File-Systeme müssen im Container vorhanden sein)
- das Host-System greift auf die Daten zu und leitet diese an den Container weiter

3.3 Sicherheit

Sicherheitsaspekte rund um den Bereich von Containern sind zu Untersuchen.

3.3.1 Root Access

Es ist gefordert, dass der Anwender auf dem Host-System keine Root-Berechtigungen besitzt und auch für das Starten des Container-Images keine weitere Berechtigungen notwendig sind. Daraus folgt, dass der Anwender ohne administrative Hilfe in der Lage sein soll, sein Container-Image zu laden und damit zu arbeiten.

In diesem Arbeitsabschnitt soll geprüft werden, welche Möglichkeiten es gibt Berechtigungen zu setzen und welche Berechtigungen notwendig sind um verschiedene Kommandos (z. B. das Laden eines Images) durchzuführen.

3.3.2 Sandboxing

Es soll untersucht werden in wie weit Container den Anwender vom Host-System abgrenzen (Hardware, Daten, andere Prozesse) und welche Ressourcen weiterhin zur Verfügung stehen.

Dem Nutzer darf keine Möglichkeit gegeben werden die Barrieren des Gast-Systems zu überschreiten und in das Host-System einzudringen oder dort Prozesse und Anwendungen zu starten. Das Host-System darf in keinem Fall von einem Anwender modifiziert werden.

3.3.3 Einschränkung der Kommunikation von verschiedenen Netzen

Es soll untersucht werden, ob es möglich ist einen geladenen Container von festgelegten Netzen abzutrennen. Zugelassene bzw. verbotene Netze werden über White- / Blacklists definiert. Verschiedene Netze sollen von Containern nicht erreichbar sein und sind aus diesem Grund nicht zugelassen.

3.4 Geschwindigkeit (Performance)

Der Einsatz von Containern ist als Vorschlag gegeben, da virtuelle Maschinen zu starke Performanceeinbußen auf großen Rechnernetzen haben.

Es soll untersucht werden, ob und wie stark die Einbußen bei der Verwendung von Container-Lösungen sind.

3.4.1 Benchmarking

In diesem Fall werden kleine Programme geschrieben, oder bereits existierende verwendet, welche die Performance des Systems prüfen sollen. Hierbei werden hauptsächlich Aspekte wie Latenz, Bandbreite und CPU-Auslastung untersucht.

3.4.2 Overhead

Es soll untersucht werden, inwiefern die Verwendung von Containern die Performance des Gesamtsystems, bzw. einzelner Knoten, senkt. Der Overhead kann aus den Resultaten des Benchmarking erkannt und analysiert werden.

3.5 Zukunftsaussichten

In diesem Arbeitspaket sind alle Aufgaben aufgeführt, welche nicht zwingend für den Projekterfolg zu bearbeiten sind. Hier sind unter anderem Ideen aufgelistet, wie das System in ferner Zukunft laufen kann.

3.5.1 User Support

Der Arbeitsschritt User Support befasst sich mit der Frage, ob ein externer Zugriff auf den Container existiert, um so dem Benutzer Support zu gewährleisten.

Mit dieser Variante ist die Möglichkeit gegeben dem Anwender bei Fragen zur Seite zu stehen und sich extern auf das System zu schalten.

Diesem Arbeitsschritt sind keine detaillierten Aufgaben untergliedert, da zum Zeitpunkt der Erstellung dieses Pflichtenheftes nicht abzusehen ist, inwiefern dieser Schritt Machbar ist und welche Variationen mit der Container-Technologie zur Verfügung stehen.

3.5.2 generic System

Unter einem generischen System ist in diesem Kontext ein System zu verstehen, welches sich automatisch den Bedürfnissen des Anwenders anpasst.

So kann z. B. der Anwender vorgeben, welche Software er verwenden möchte und das System erstellt dazu automatisch ein Image, mit benötigtem Betriebssystemen, notwendigen Bibliotheken und der zu nutzenden Software (in geforderter Version), welches dem Anwender zur Verfügung gestellt wird bzw. automatisch geladen werden kann.

3.5.3 Orchestrierung

Es soll untersucht werden, wie das Image im Gesamtsystem geladen werden kann. Darüber hinaus sind Aspekte der Hardwareverteilung (z. B. Anfordern von mehreren Cores) zu Berücksichtigen.

Als ein Vorschlag ist die Verwendung von OpenStack als Orchestrierungstool gegeben, welcher zu untersuchen ist.

4 Projektplan

Der hier dargestellte Projektplan zeigt die Reihenfolge der Aufgaben und deren Zuordnung in das Arbeitspaket. Nach Abschluss eines Arbeitspaketes ist der dazu gehörige Meilenstein als Teilabschluss vorgesehen.

1. Container & Infiniband (Machbarkeit)

2. Erstellen eines Images (Machbarkeit)

3. Shared File System (Machbarkeit)

4. **Meilenstein Machbarkeitsprüfung**

5. Benchmarking (Performance)

6. Overload (Performance)

7. **Meilenstein Benchmarking**

8. Root Access (Sicherheit)

9. Sandboxing (Sicherheit)

10. Einschränkung der Kommunikation von verschiedenen Netzen (Sicherheit)

11. **Meilenstein Sicherheit**

12. Projektabschluss & Zusammenfassung

13. **Meilenstein Projektabschluss**

14. User Support (Aussichten)

15. Orchestrierung (Aussichten)

16. generic System (Aussichten)

5 Zeitplanung

Da das Projekt zusätzlich an der Hochschule für Technik und Wirtschaft in Berlin (HTW-Berlin) in einigen Kursen als Abgabe vorgesehen ist, sind die Zeitvorgaben der Hochschule einzuhalten.

Der Erste Projektabschnitt Machbarkeit wird im Rahmen des Kurses *Forschungsprojekt 1* absolviert.

Aufgaben, welche laut Projektplan nach dem Meilenstein Machbarkeitsprüfung geplant sind, werden im Kurs *Forschungsprojekt 2* oder der danach anstehenden Masterarbeit behandelt.

Darüber hinaus werden grundlegende Begrifflichkeiten und Zusammenhänge in einem separaten Dokument für den Kurs *Independent Coursework* beschrieben. Innerhalb dieser Ausarbeitung werden auch verschiedene Lösungen aufgezeigt und deren Vor- und Nachteile beschrieben.

Wichtig: Dieser Zeitentwurf ist lediglich eine Richtlinie und ist nicht zwingend einzuhalten. Durch stetige Kommunikation mit dem DESY und Prof. Dr. Hermann Heßling (Zuständiger Professor der Hochschule) können Projektaufgaben verschoben, entfernt oder neue Aufgaben hinzugefügt werden.