Freie Universität Berlin

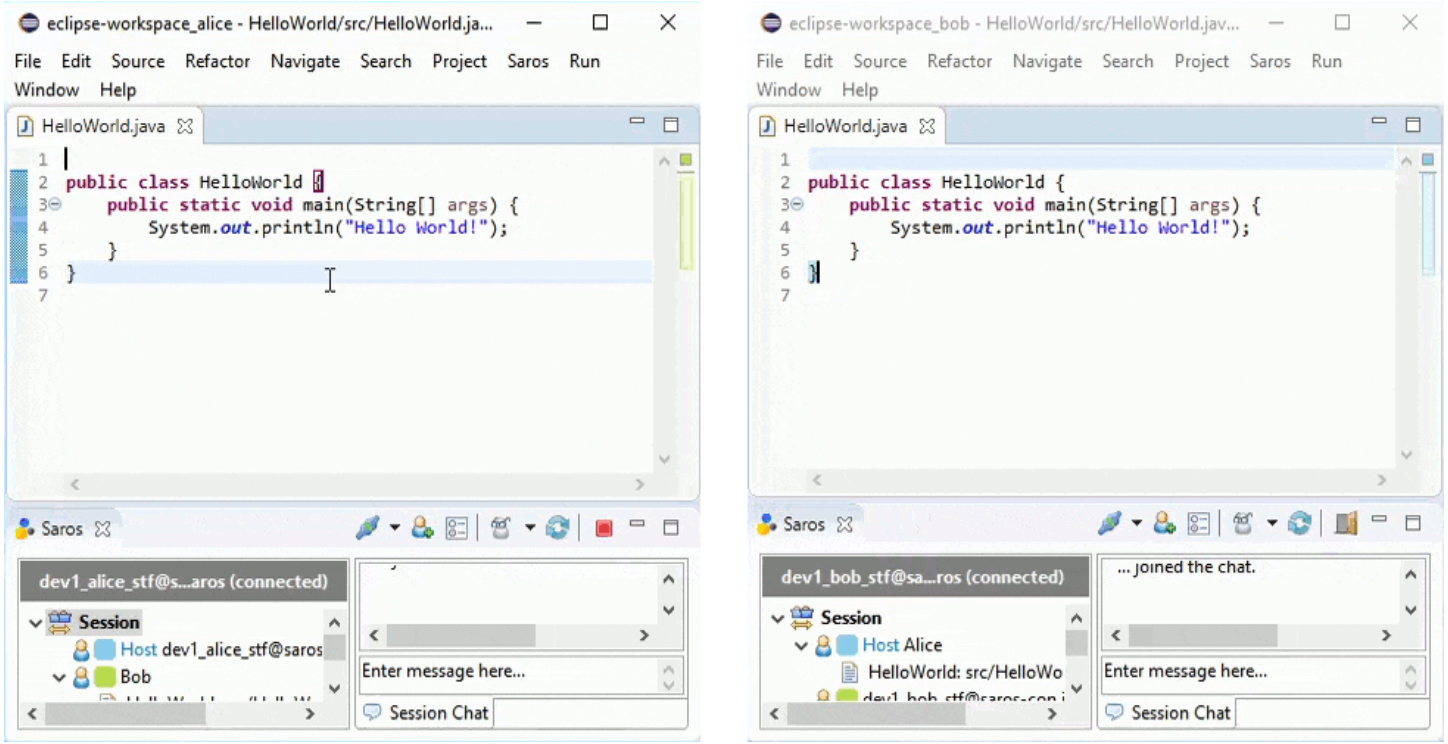
Verteidigung Masterarbeit

Analyse und Bewertung einer Erweiterung des Tools Saros für den Editor Visual Studio Code und Validierung konzeptioneller Optimierungen

Gliederung

1. Einleitung
2. Analyse
3. Lösungsansätze
4. Fazit
5. Ausblick

Was ist Saros?



Quelle: https://www.saros-project.org/assets/images/animation/saros.gif

Ausgangssituation

- Erster Prototyp für VSCode in Bachelorarbeit von Michael Schäfer implementiert

- Ziel: Implementierung der grundlegendsten Funktionen

Freie Universität Berlin

Aufgabenstellung

Analyse und Bewertung einer Erweiterung des Tools Saros für den Editor Visual Studio Code und Validierung konzeptioneller Optimierungen

Erste Analyse

- Erste manuelle Tests bestätigten die Nutzbarkeit der wichtigsten Kernfunktionalitäten

- Zusätzlich wurde jedoch ein grundlegendes Problem in der Architektur identifiziert

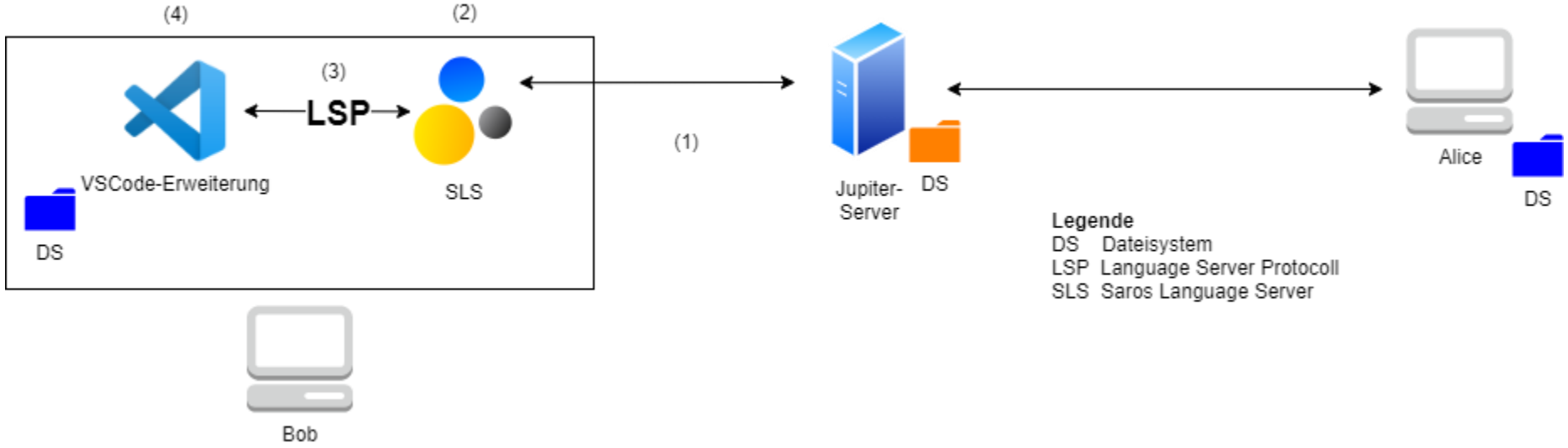
Das Problem mit der Asynchronität

- Abkapselung von Erweiterungen in VSCode

- Kommunikation über asynchrone Extension API

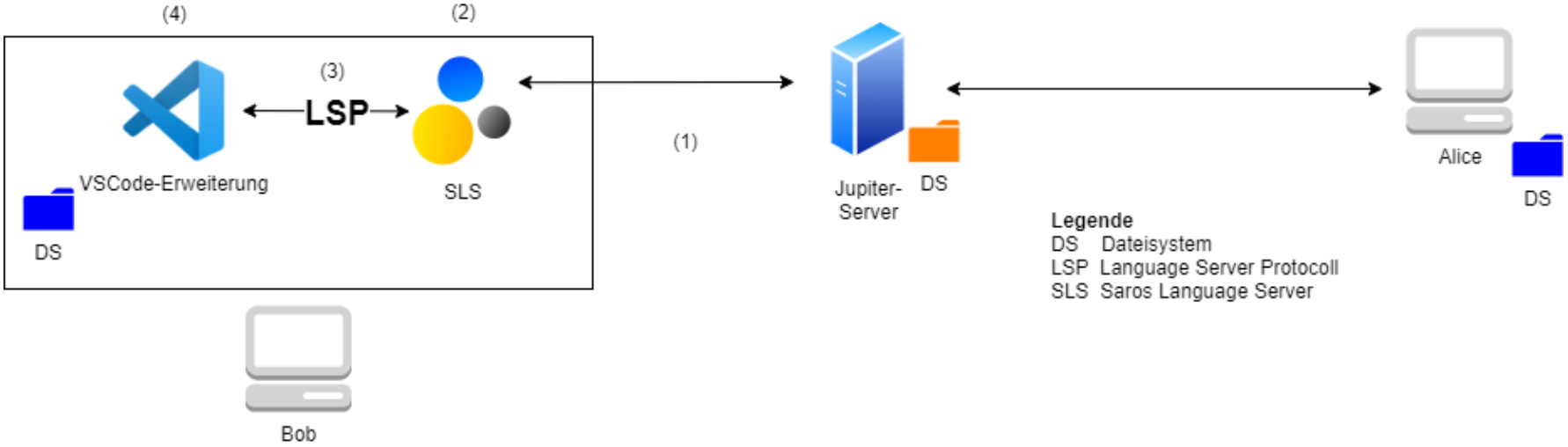
Freie Universität Berlin

Problembeispiel



Freie Universität Berlin

Problembeispiel



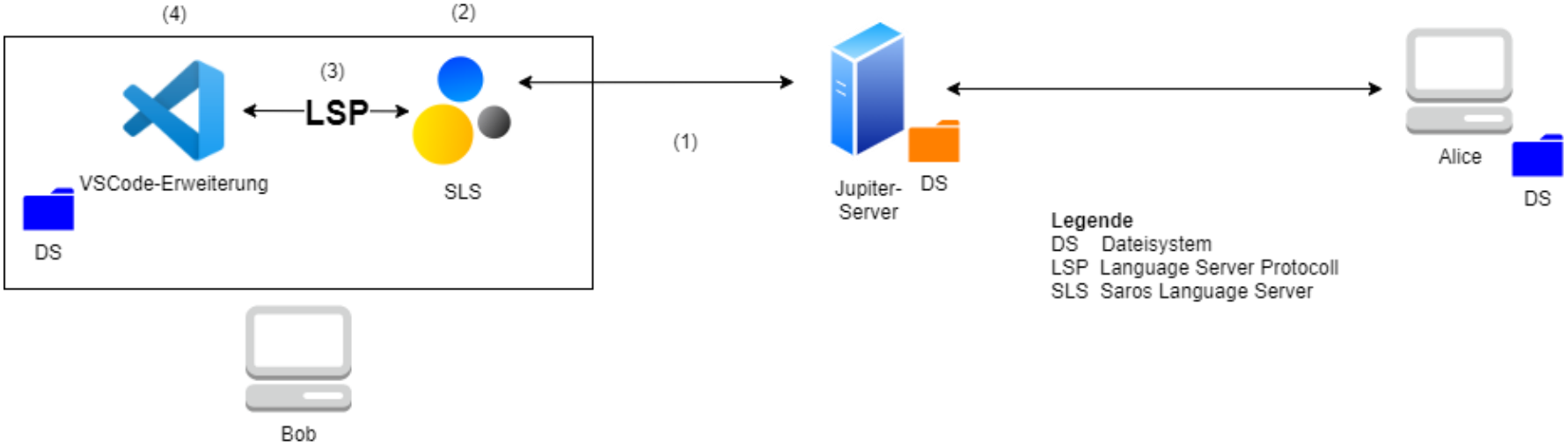
Hello world

Hello world

Hello world

Freie Universität Berlin

Problembeispiel

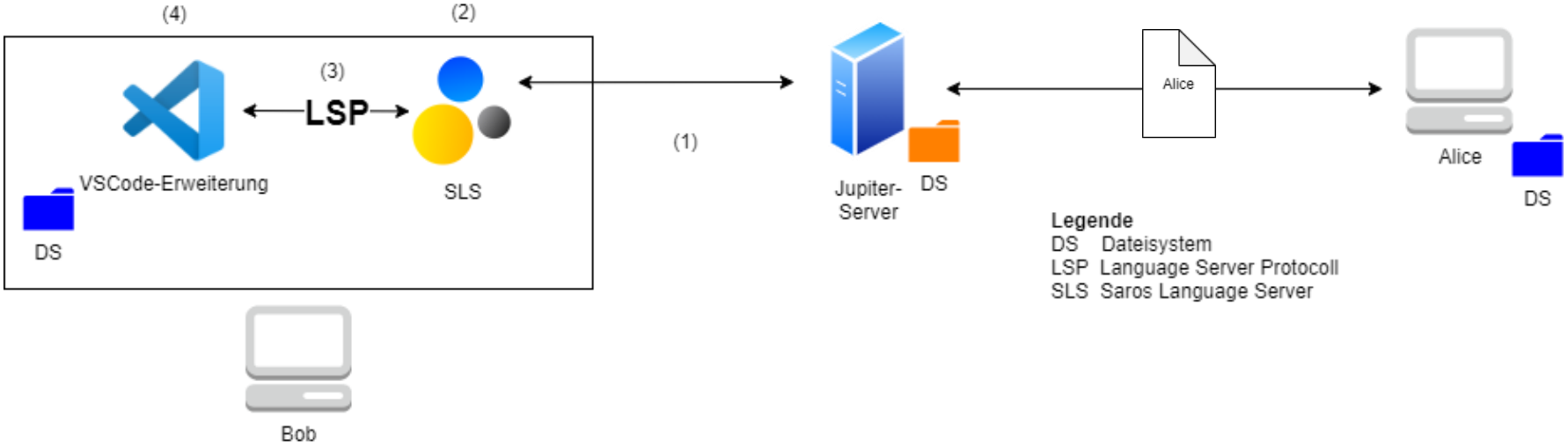


Hello world

Hello world

Hello my unique world

Problembeispiel



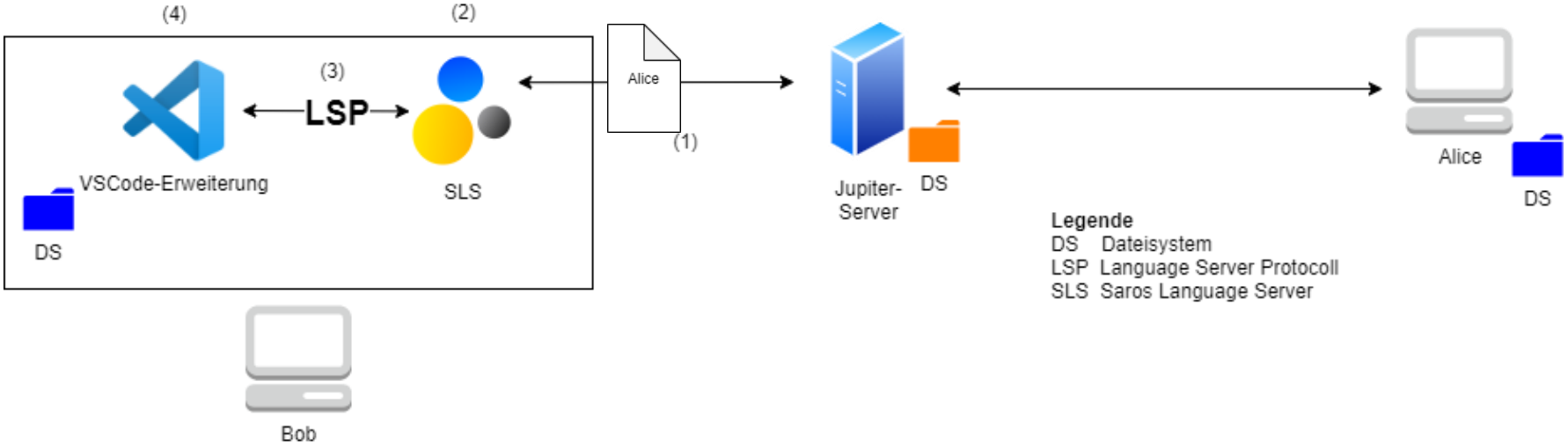
Hello world

Hello world

Hello my unique world

Freie Universität Berlin

Problembeispiel



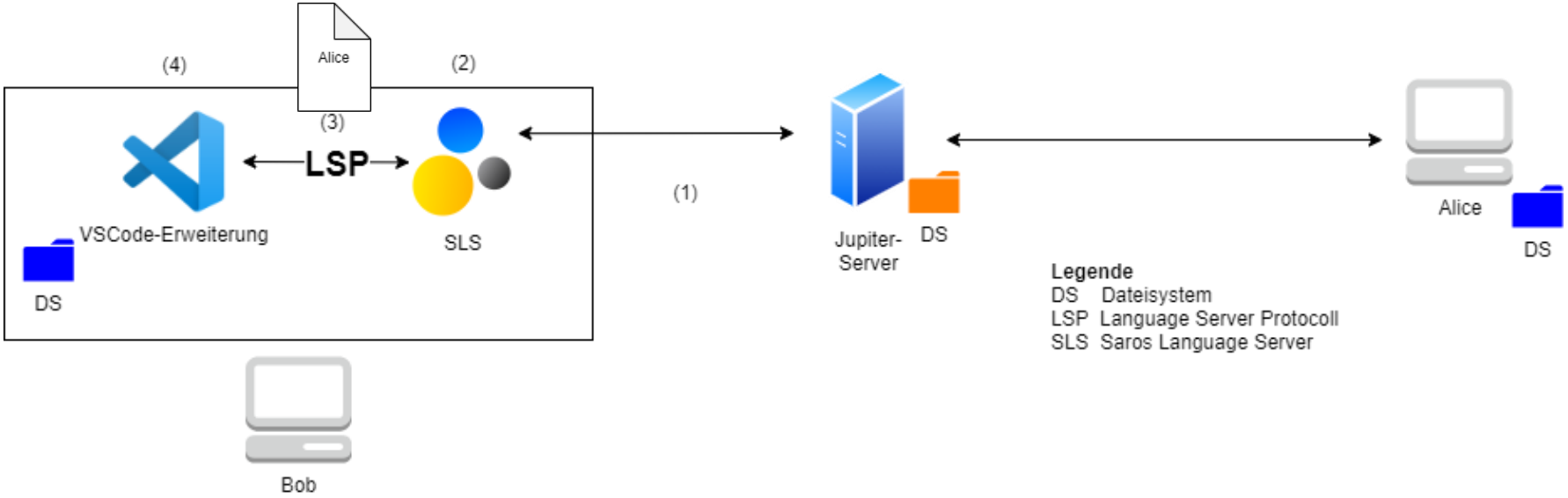
Hello world

Hello my unique world

Hello my unique world

Freie Universität Berlin

Problembeispiel



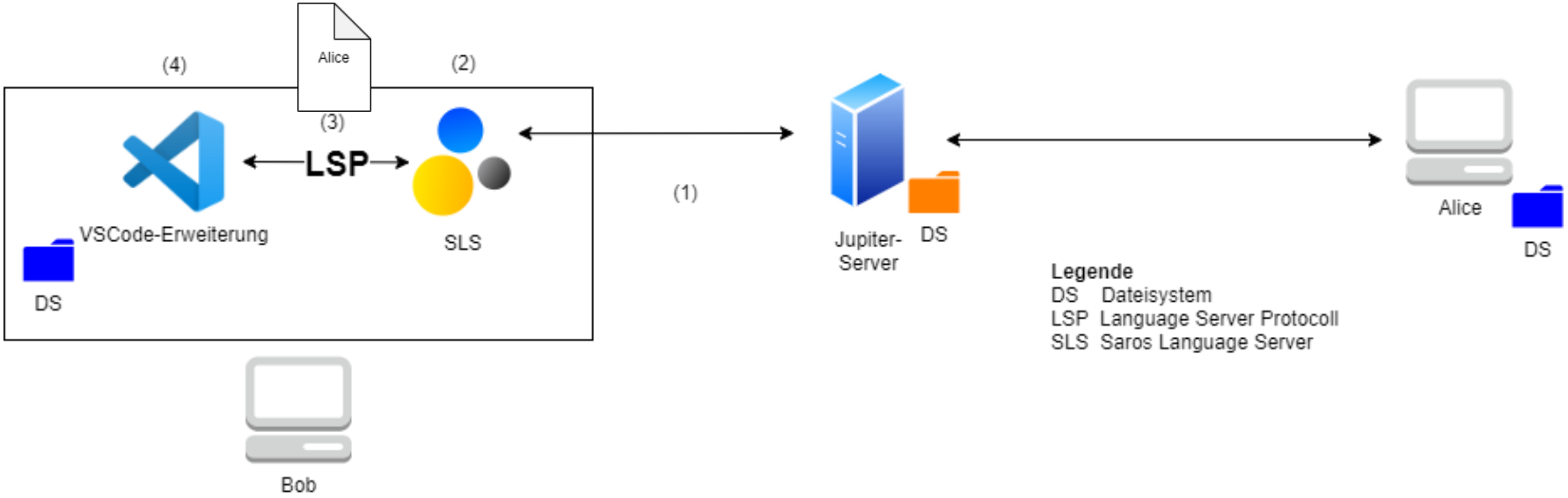
Hello world

Hello my unique world

Hello my unique world

Freie Universität Berlin

Problembeispiel



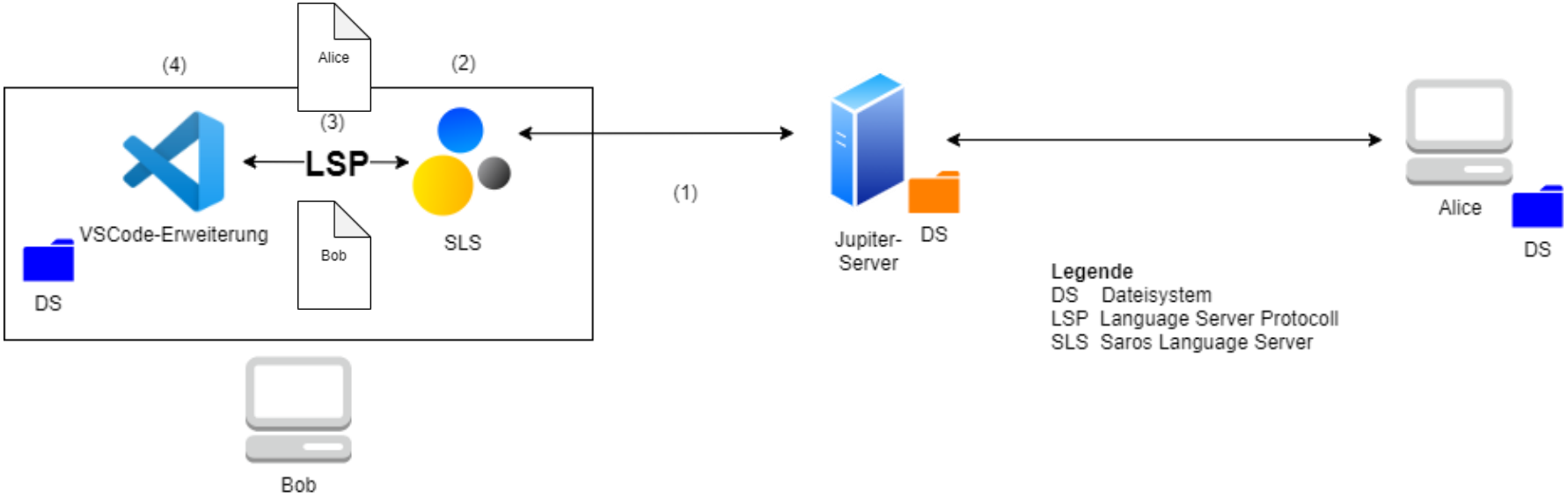
Hello world!

Hello my unique world

Hello my unique world

Freie Universität Berlin

Problembeispiel



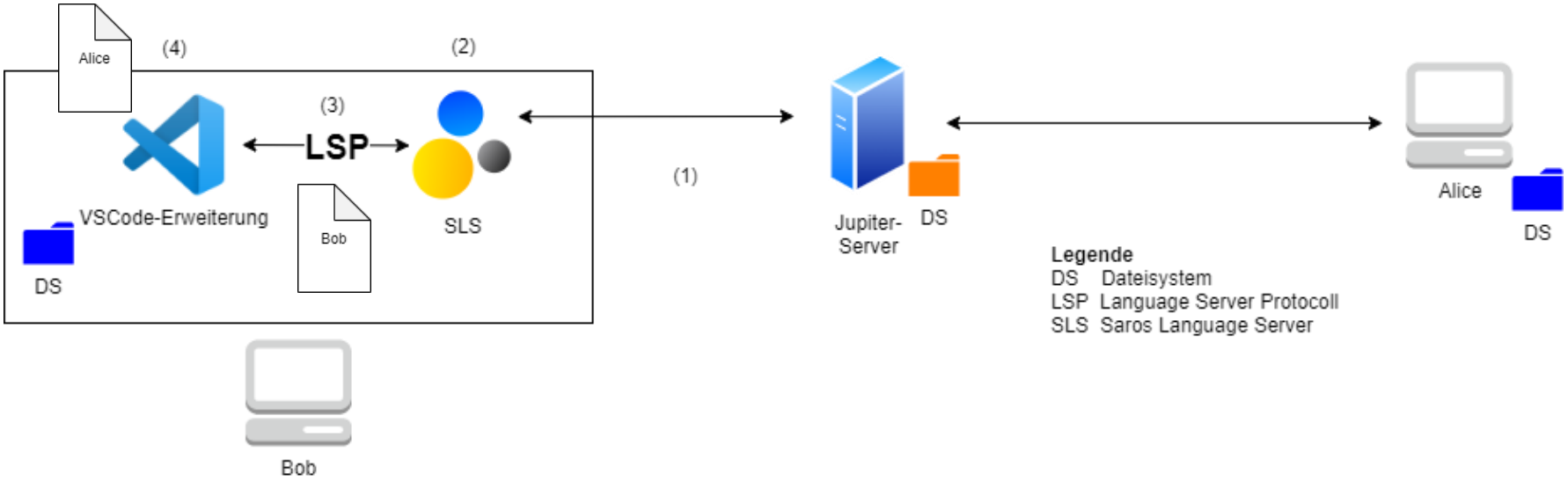
Hello world!

Hello my unique world

Hello my unique world

Freie Universität Berlin

Problembeispiel



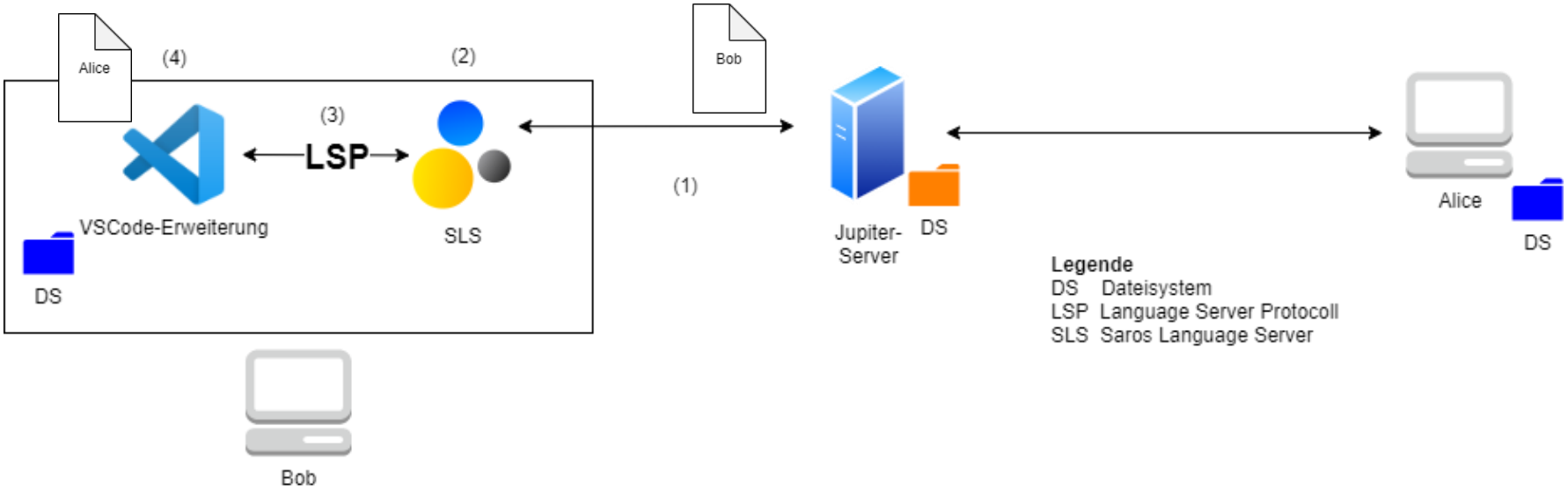
Hello my unique world!

Hello my unique world

Hello my unique world

Freie Universität Berlin

Problembeispiel



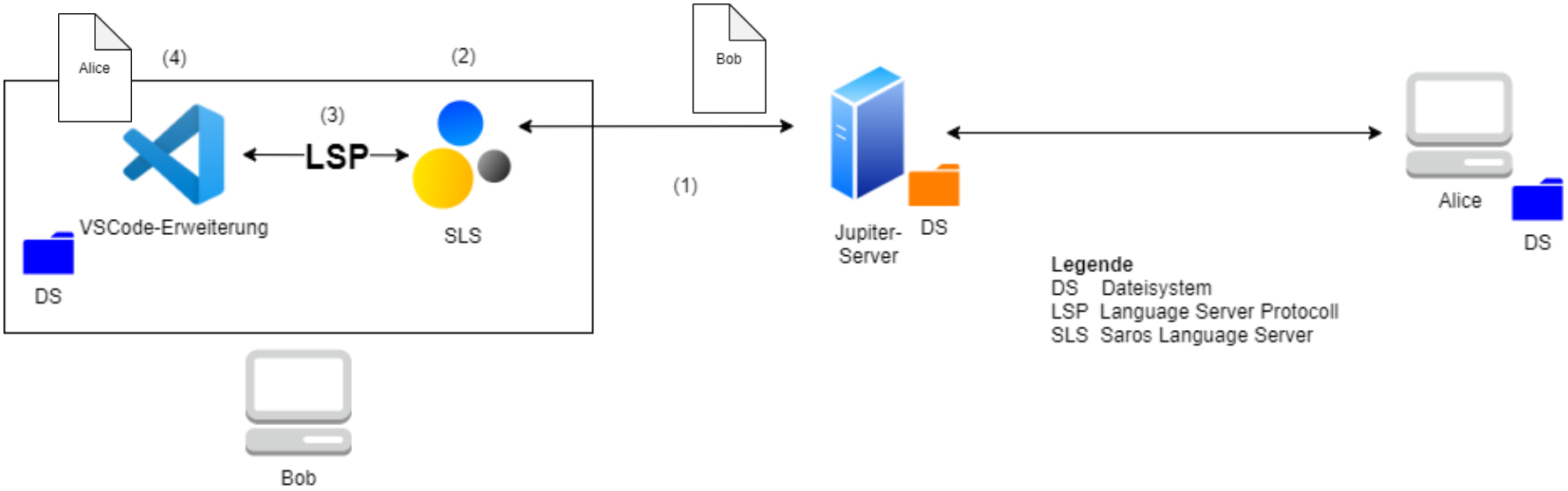
Hello my unique world!

Hello my unique world

Hello my unique world

Freie Universität Berlin

Problembeispiel



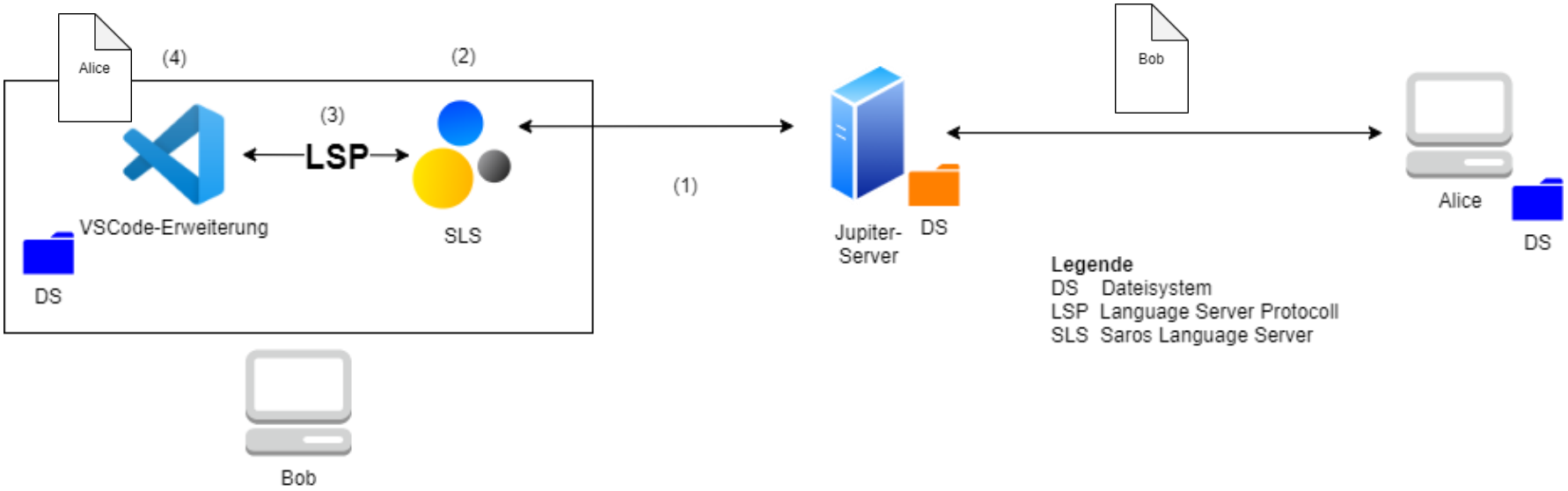
Hello my unique world!

Hello my un!ique world

Hello my unique world

Freie Universität Berlin

Problembeispiel

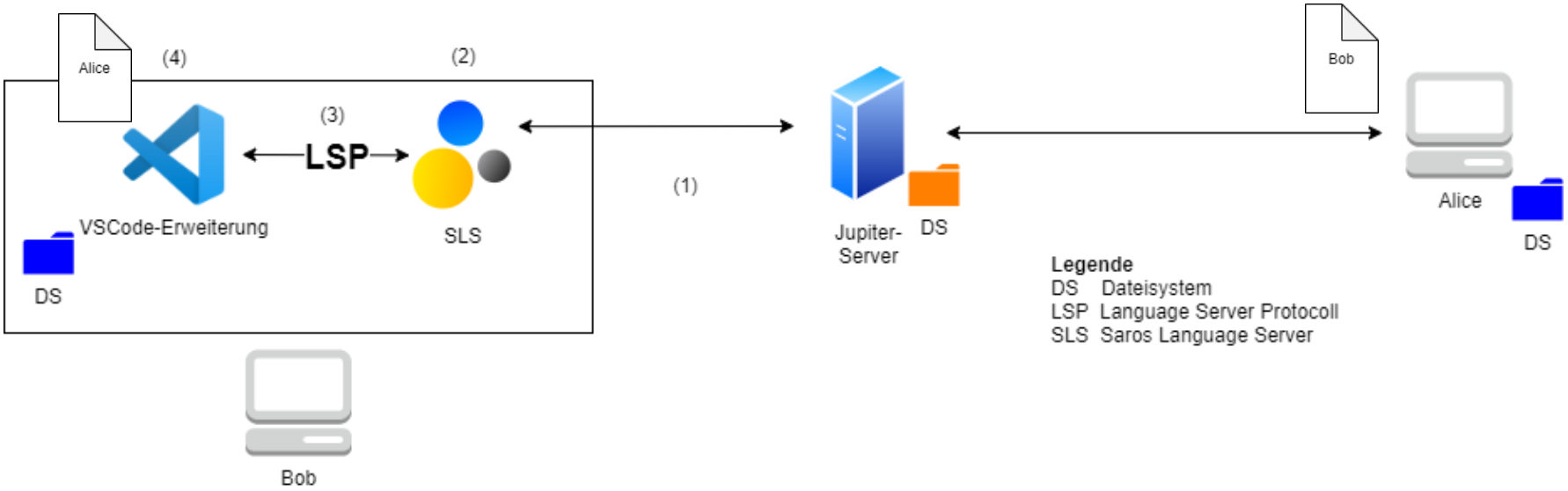


Hello my unique world!

Hello my un!ique world

Hello my unique world

Problembeispiel



Hello my unique world!

Hello my un!ique world

Hello my un!ique world

Anforderungen

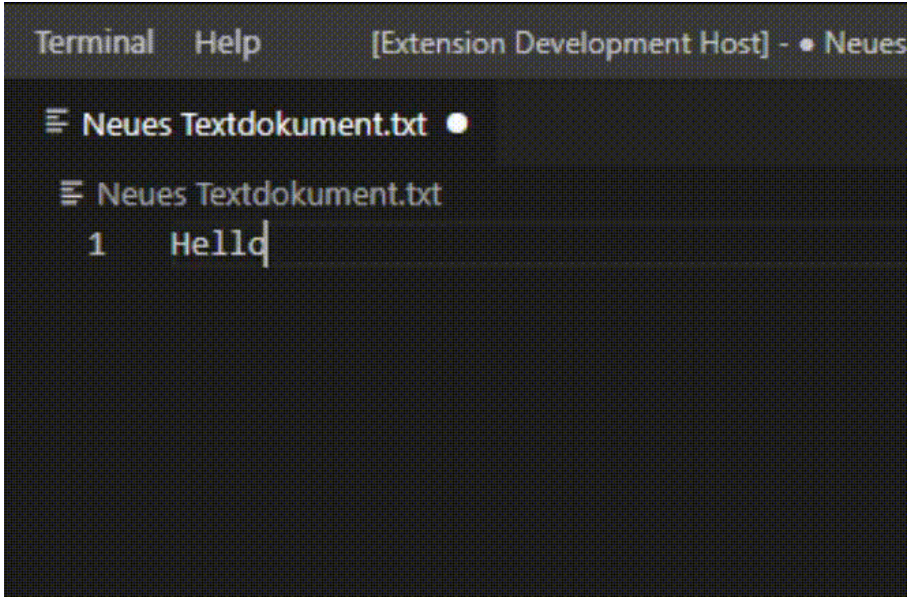
1. Der Problemfall muss gelöst werden

2. Der SLS soll erhalten bleiben

3. Die Nutzererfahrung muss mit anderen Implementierungen von Saros vergleichbar und nachvollziehbar sein

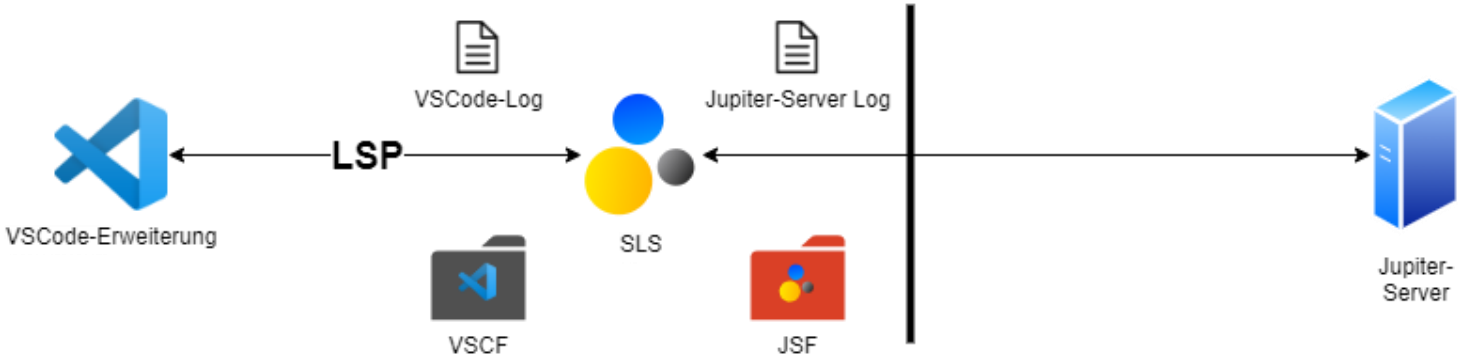
Konzept 1

- Sofortiges rückgängig machen von Nutzeränderungen im Saros-VSCode Klienten
- Anschließendes Einfügen nach Verarbeitung der Änderung durch den SLS



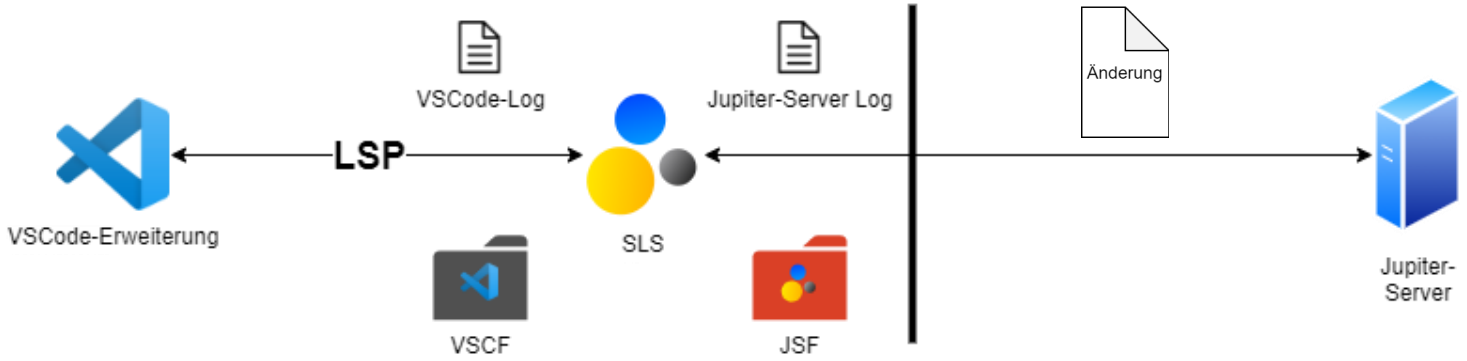
Konzept 2

- Speichern der unterschiedlichen Datenstände in Filesystemen im SLS
- Zwischenspeichern aller ausgehenden Nachrichten vom SLS in Log-Dateien



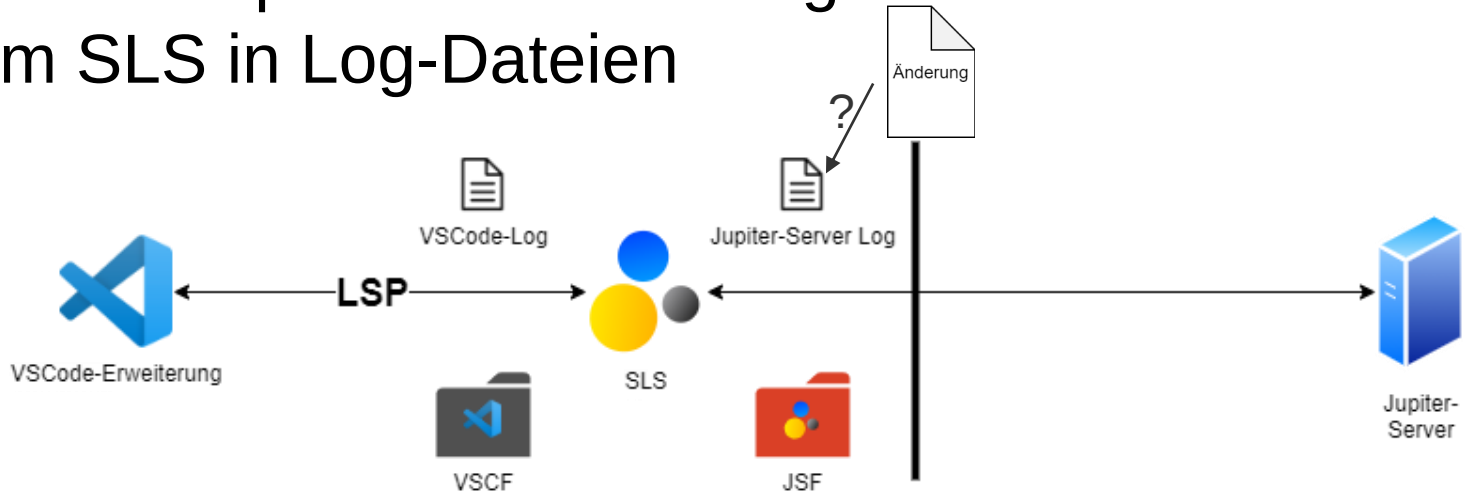
Konzept 2

- Speichern der unterschiedlichen Datenstände in Filesystemen im SLS
- Zwischenspeichern aller ausgehenden Nachrichten vom SLS in Log-Dateien



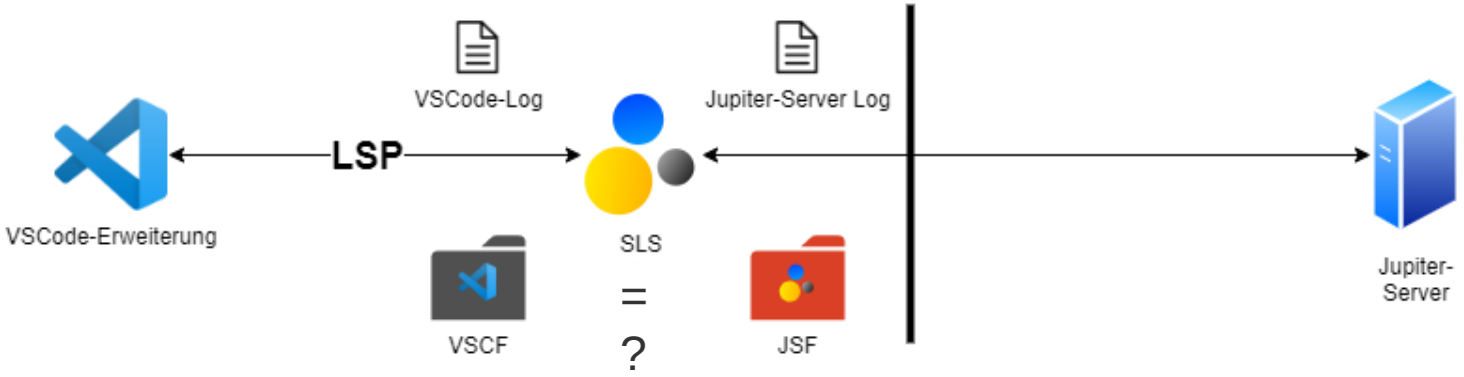
Konzept 2

- Speichern der unterschiedlichen Datenstände in Filesystemen im SLS
- Zwischenspeichern aller ausgehenden Nachrichten vom SLS in Log-Dateien



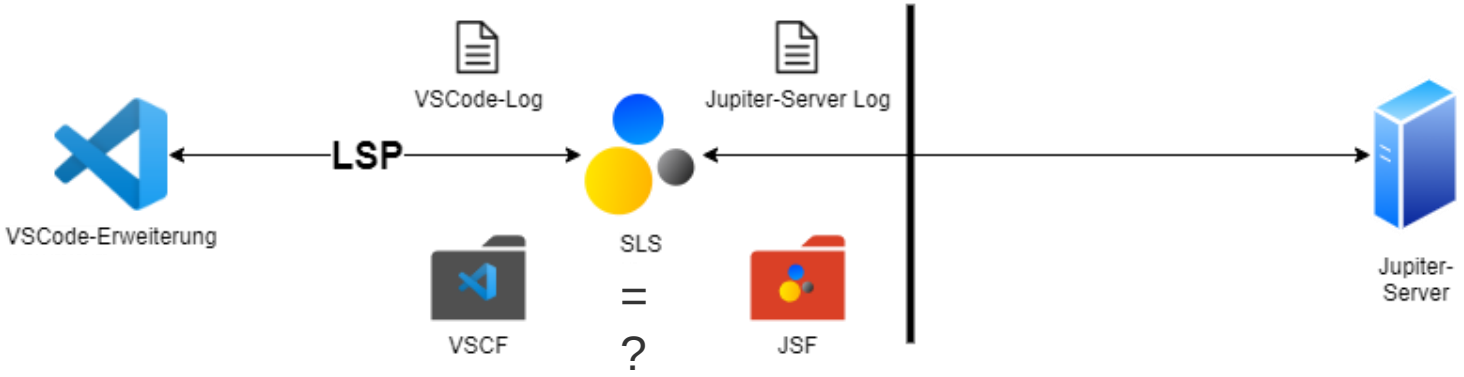
Konzept 2

- Speichern der unterschiedlichen Datenstände in Filesystemen im SLS
- Zwischenspeichern aller ausgehenden Nachrichten vom SLS in Log-Dateien



Konzept 2

- Hohe Komplexität
- Umbau des Jupiter-Servers notwendig
- Aufwendig in der Wartung



Änderung der Aufgabenstellung

- Verschieben des Fokus der Arbeit komplett auf die Lösung dieses Problems

- Analyse von bestehenden Konzeptes zur Konfliktlösung zwischen SLS und Saros-VSCode Klient als Grundlage eines neuen Konzeptes

Vergleich bestehender Konzepte

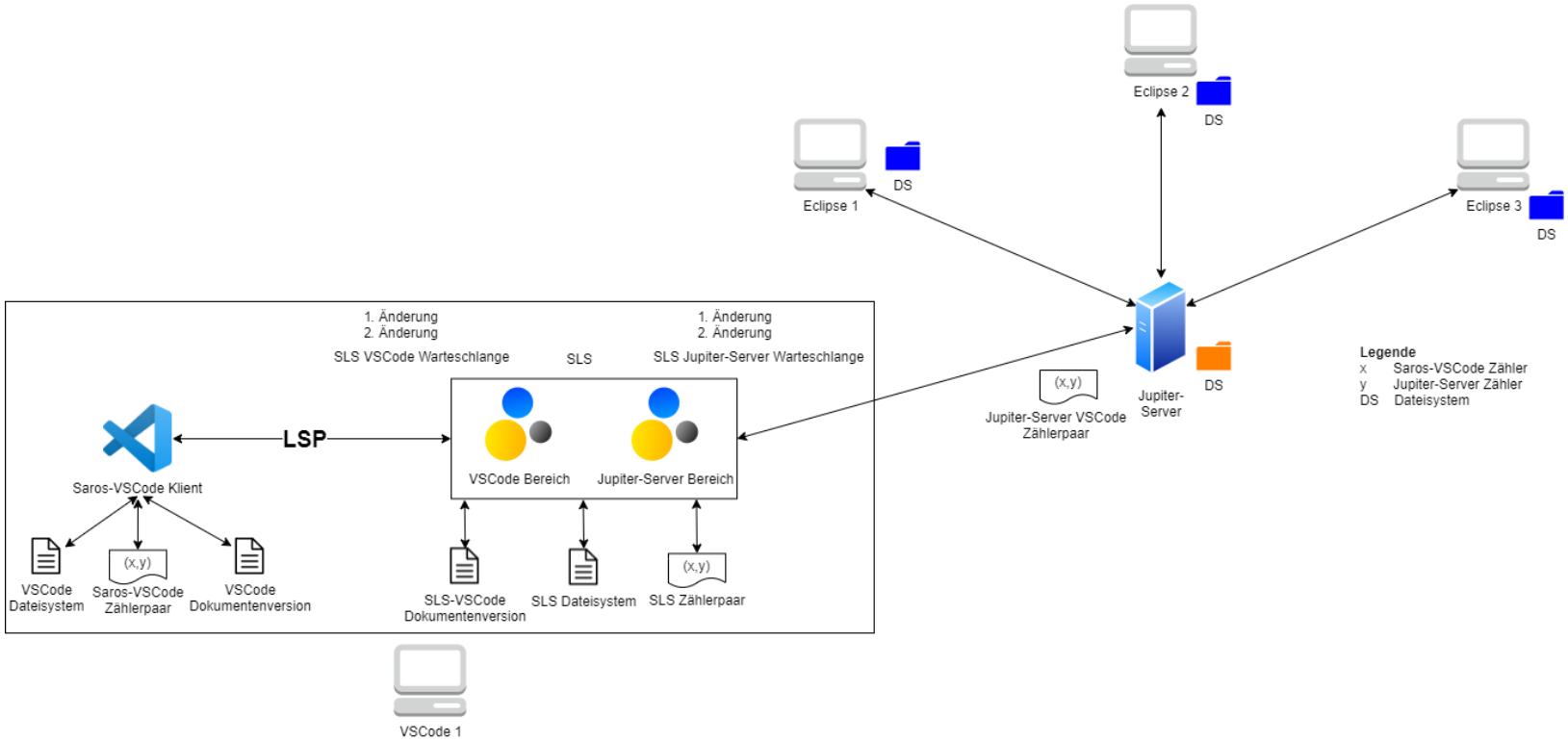
	Verteilte Datenbanksysteme	CRDTs	OT
Vorteile	- Viele ausgereifte Konzepte	- Gut erforschte, moderne Alternative - Löst viele Probleme mit OT - Praktische Beispiele in kollaborativen Editoren vorhanden	- Gleichen Funktionen wie CRDTs - Bereits in Saros implementiert - Vorhandenes Know How
Nachteile	- Zu wenig Informationen über die Funktionsweise in der Praxis	- Neue Technologie im Projekt Saros - Fehlendes Know How im Saros-Team	

Finale Lösung - Anforderungen

1. Der Problemfall muss gelöst werden
2. Der SLS soll erhalten bleiben
3. Die Nutzererfahrung muss mit anderen Implementierungen von Saros vergleichbar und nachvollziehbar sein
4. Soll zur Konfliktauflösung operational transformation verwenden
5. Regeln zur Konfliktbehandlung von Jupiter müssen eingehalten werden

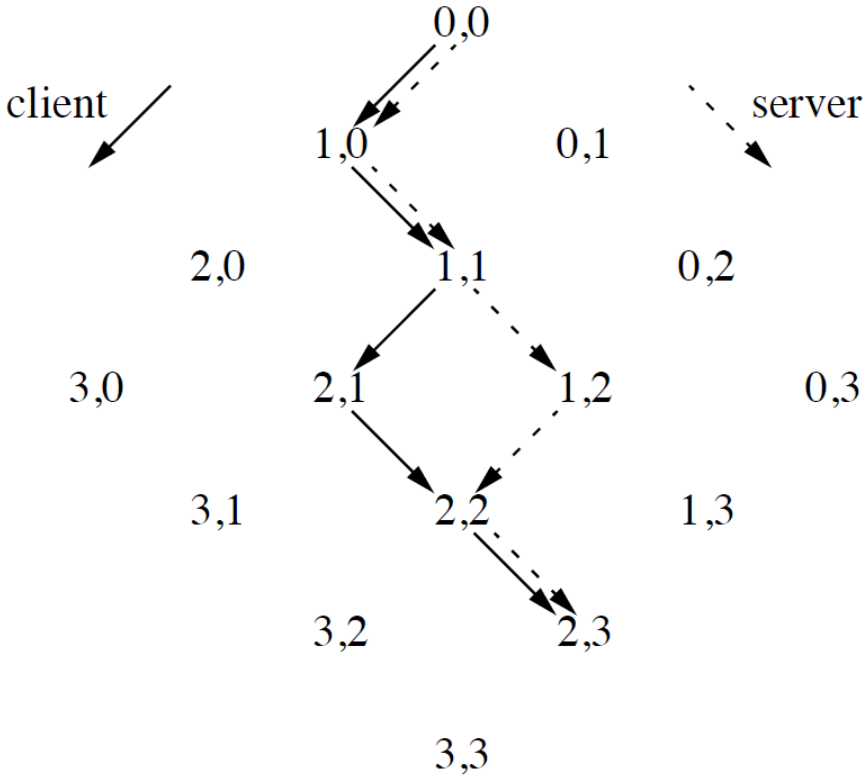
Freie Universität Berlin

Finale Lösung - Architektur

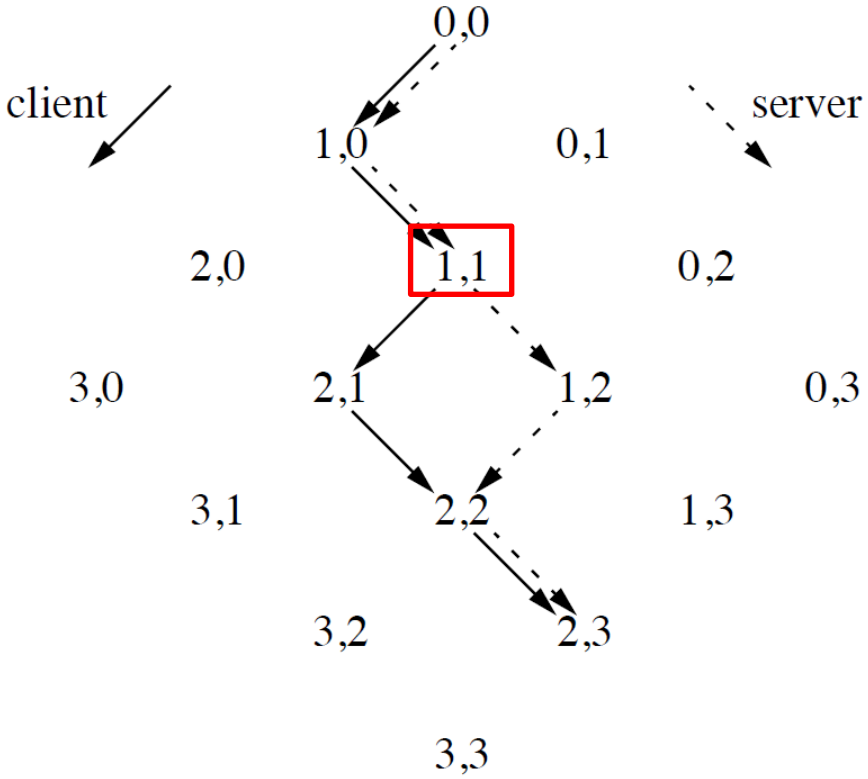


Freie Universität Berlin

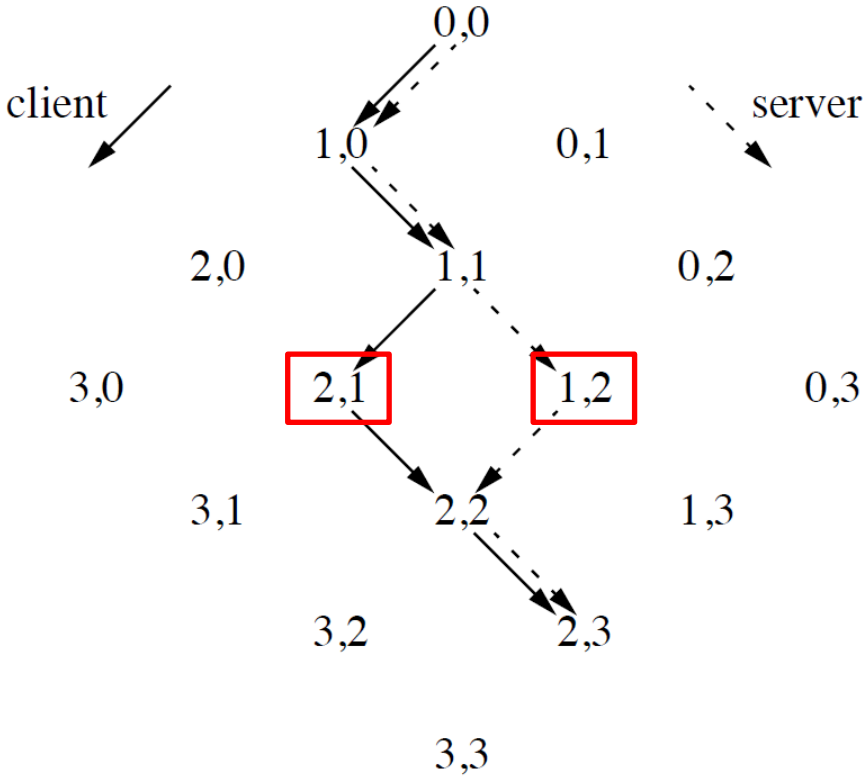
Finale Lösung - Architektur



Finale Lösung - Architektur

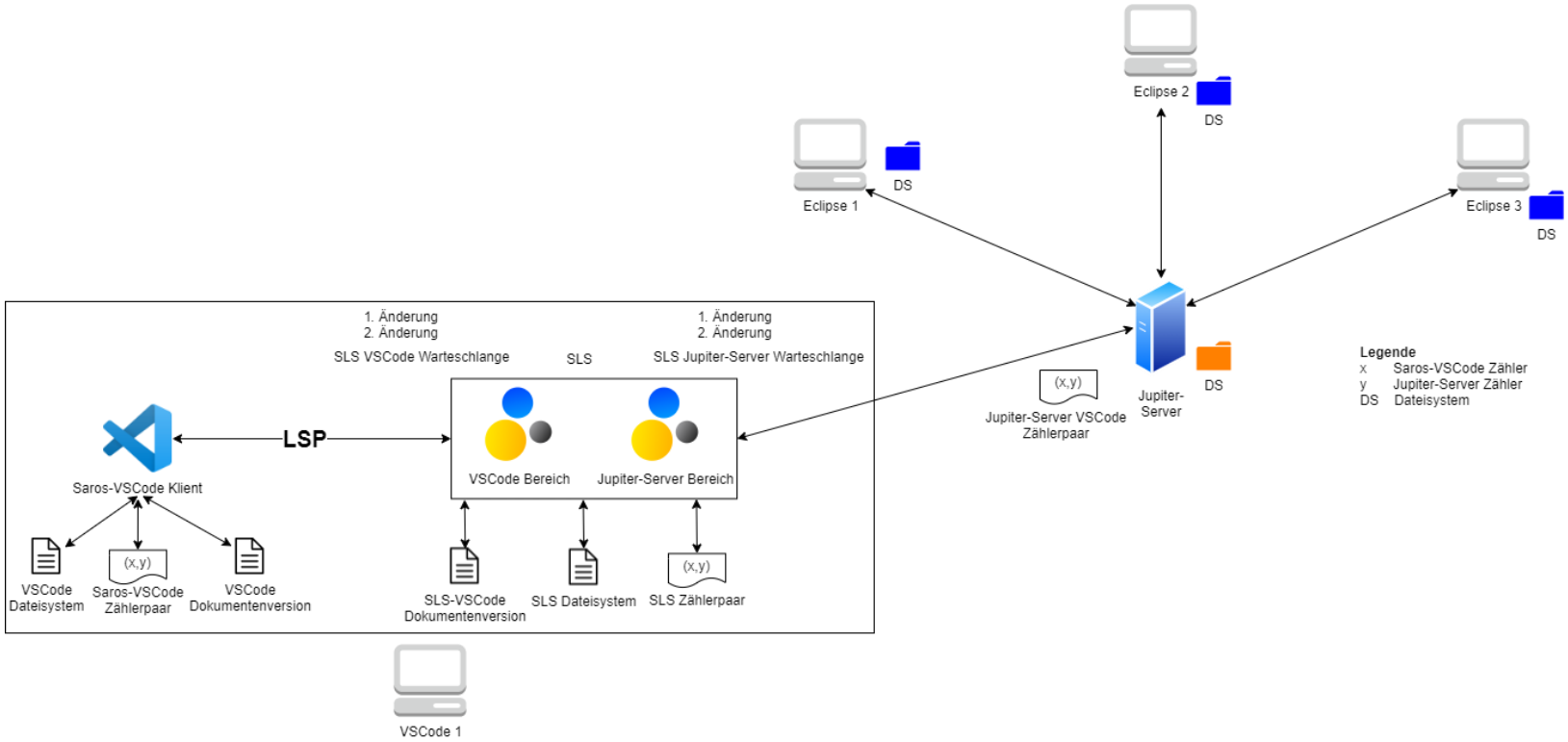


Finale Lösung - Architektur



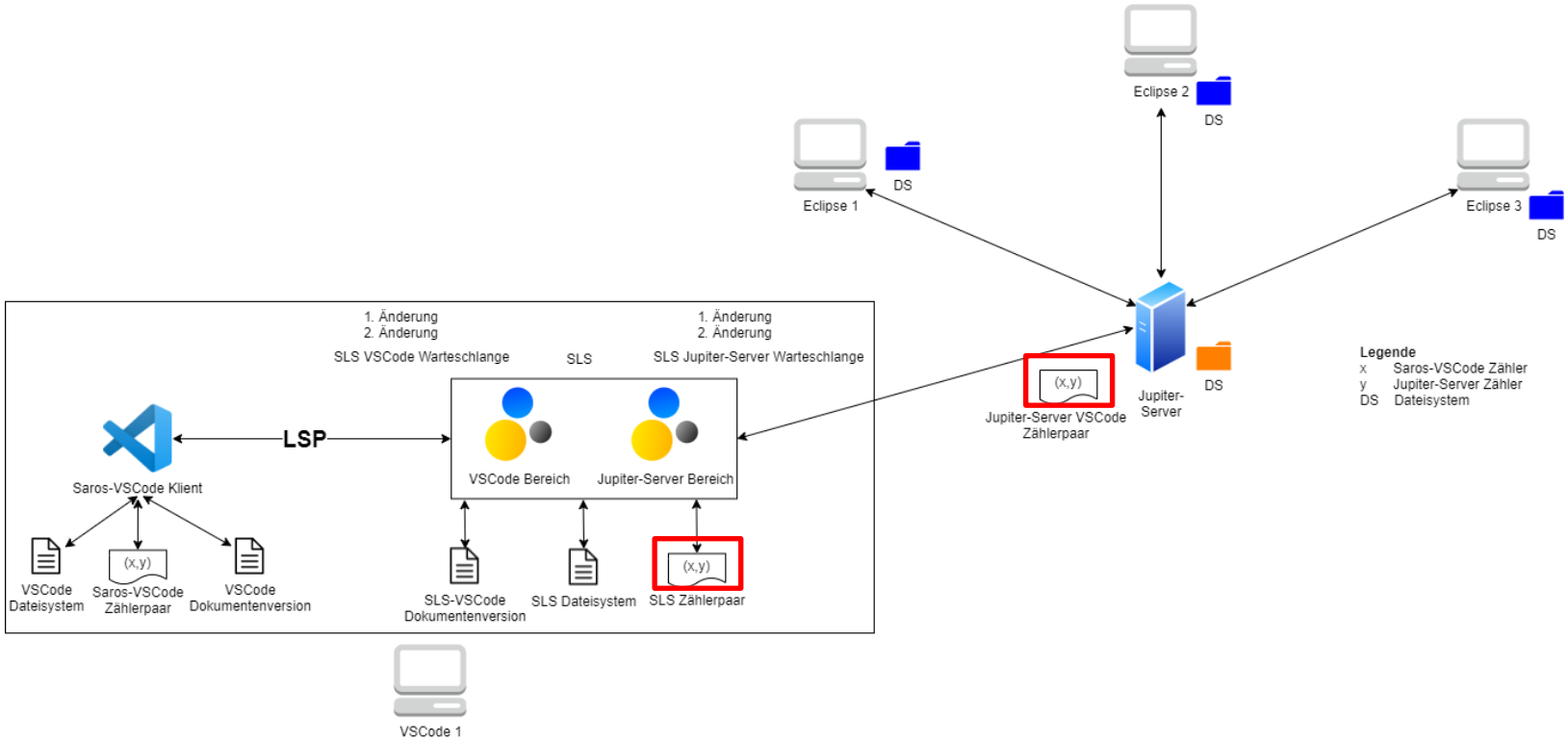
Freie Universität Berlin

Finale Lösung - Architektur



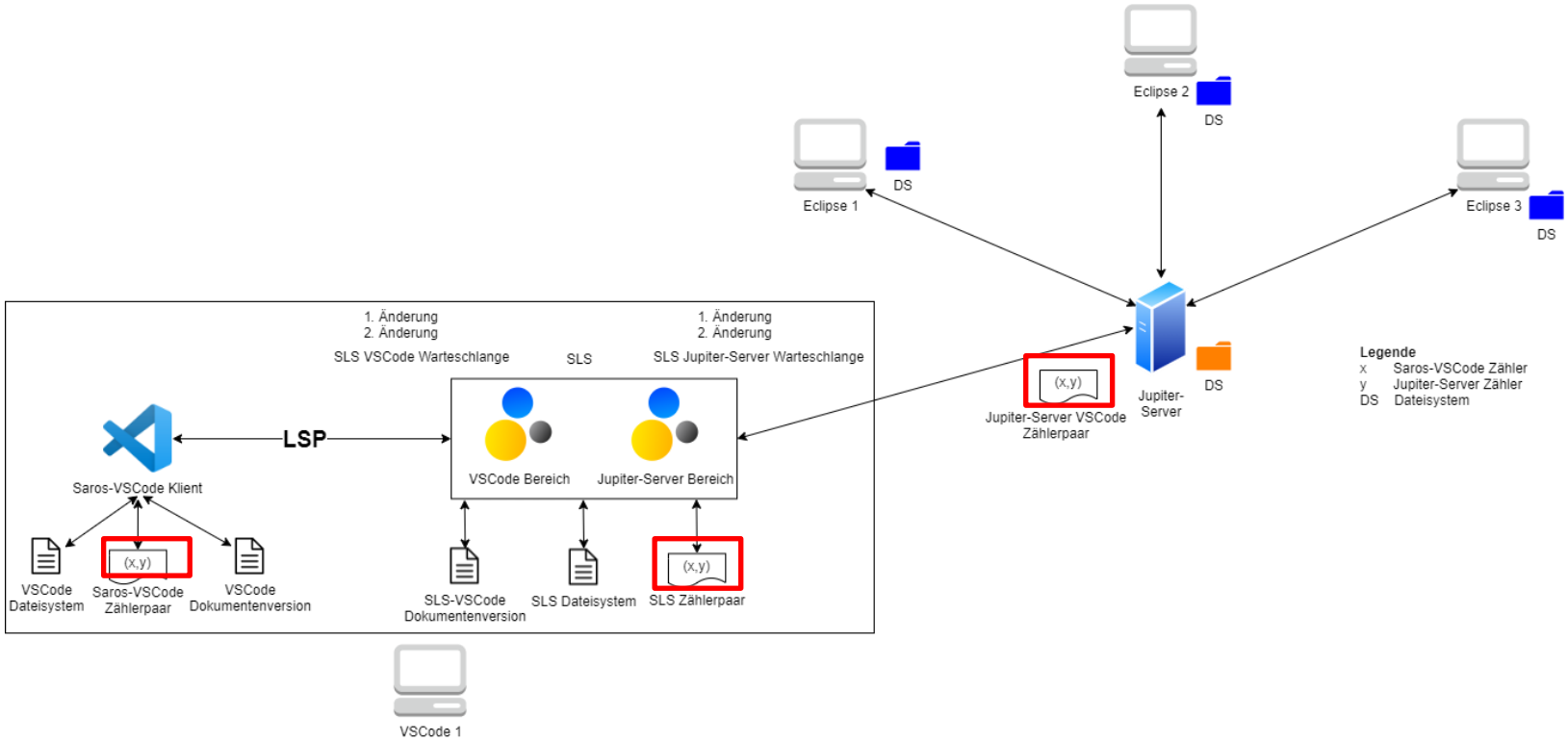
Freie Universität Berlin

Finale Lösung - Architektur



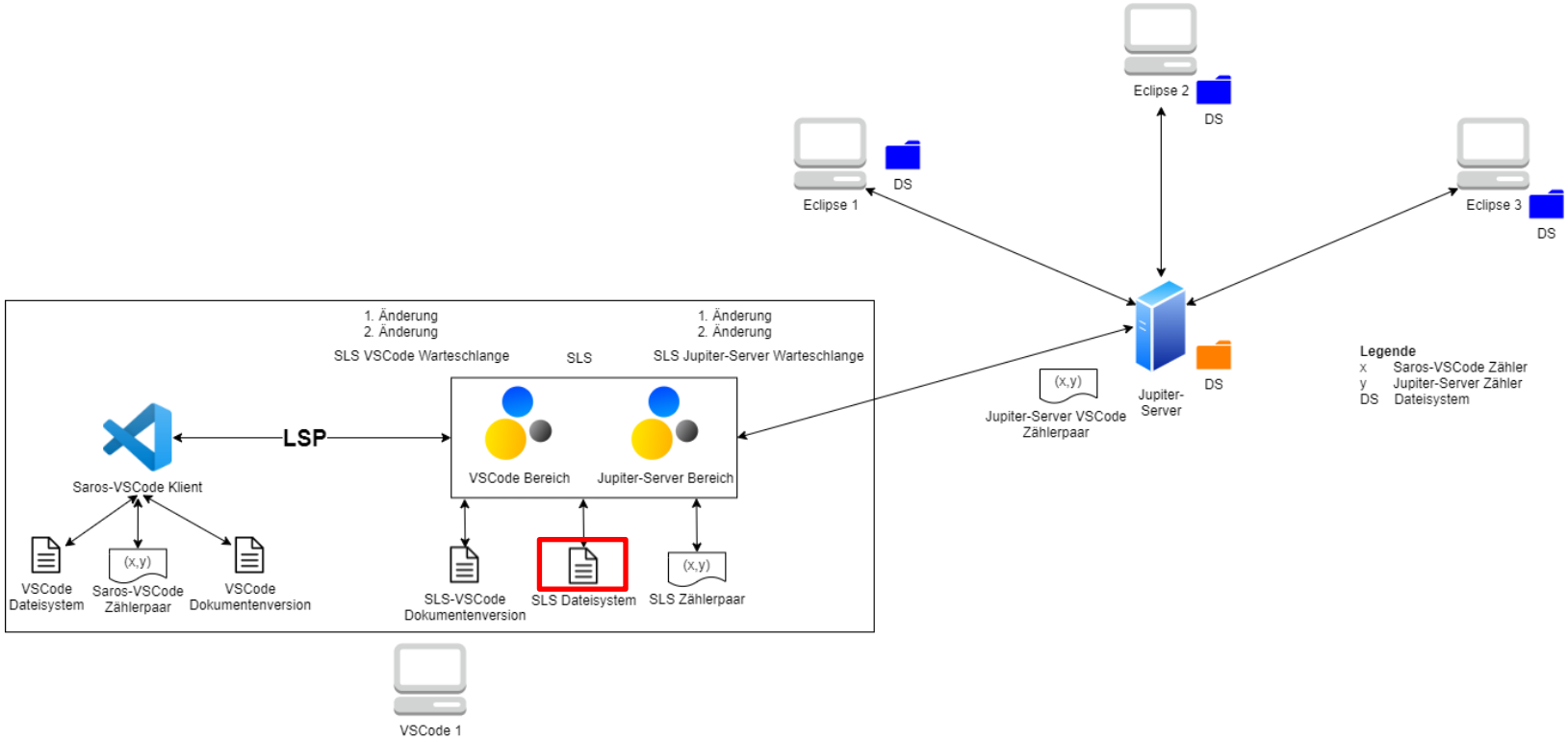
Freie Universität Berlin

Finale Lösung - Architektur



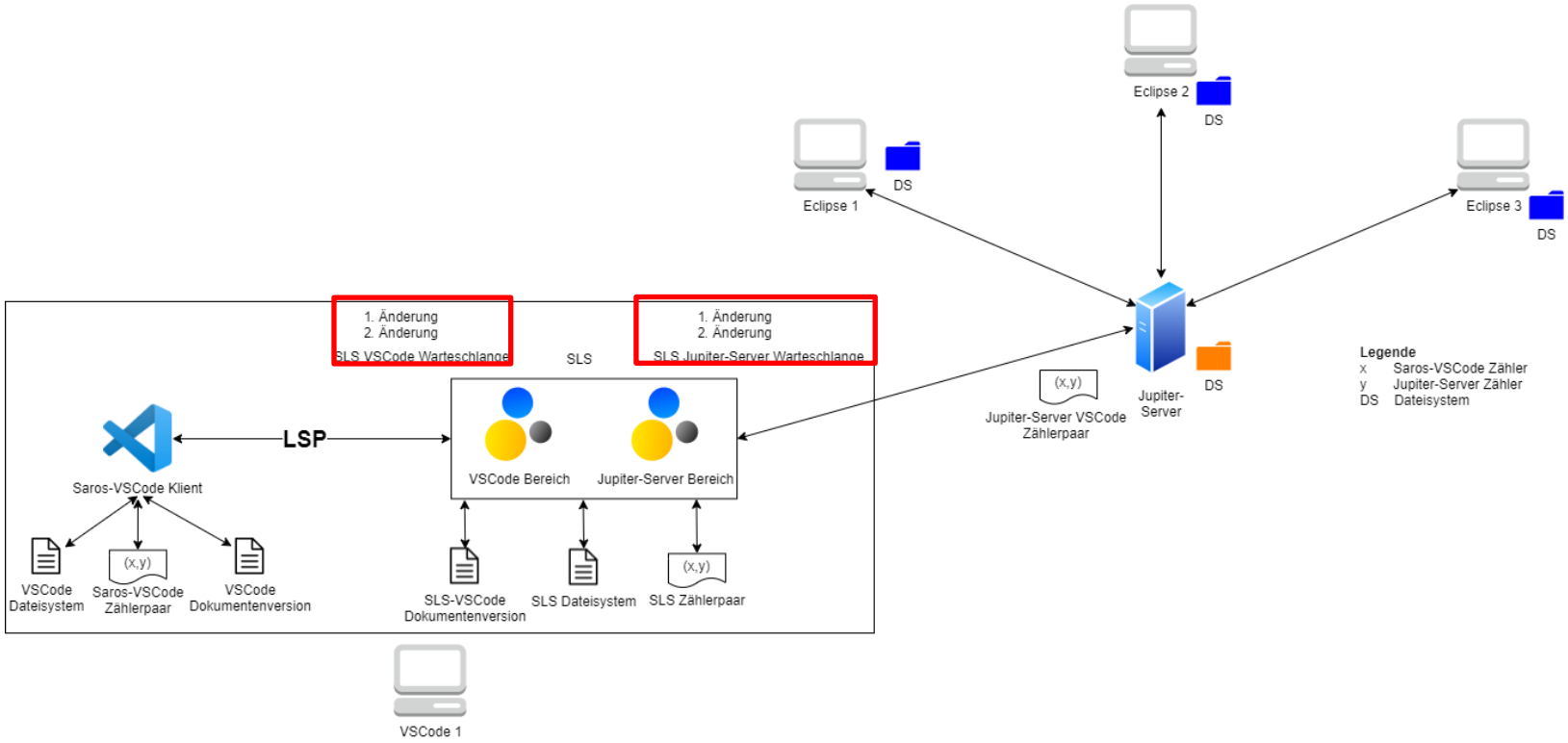
Freie Universität Berlin

Finale Lösung - Architektur



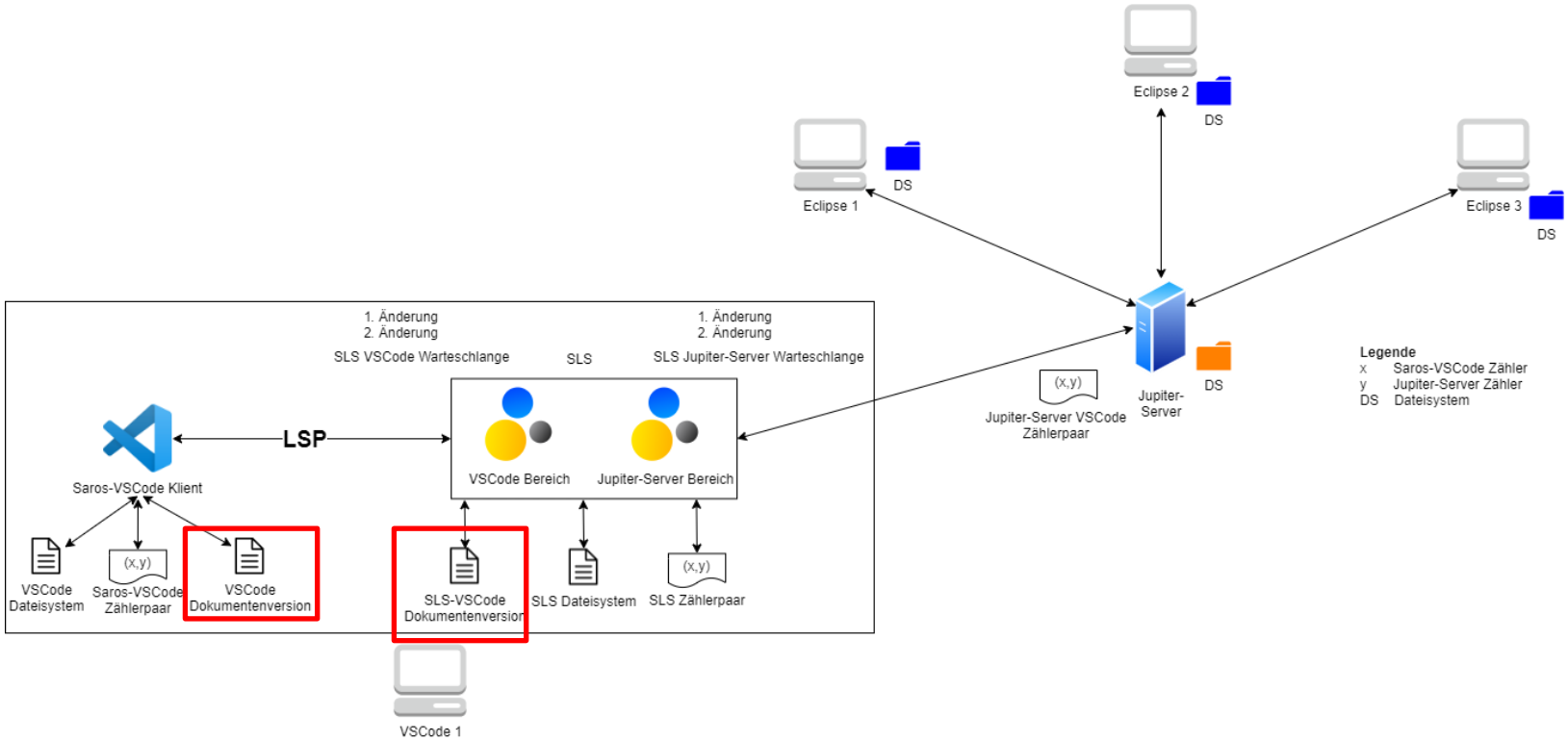
Freie Universität Berlin

Finale Lösung - Architektur

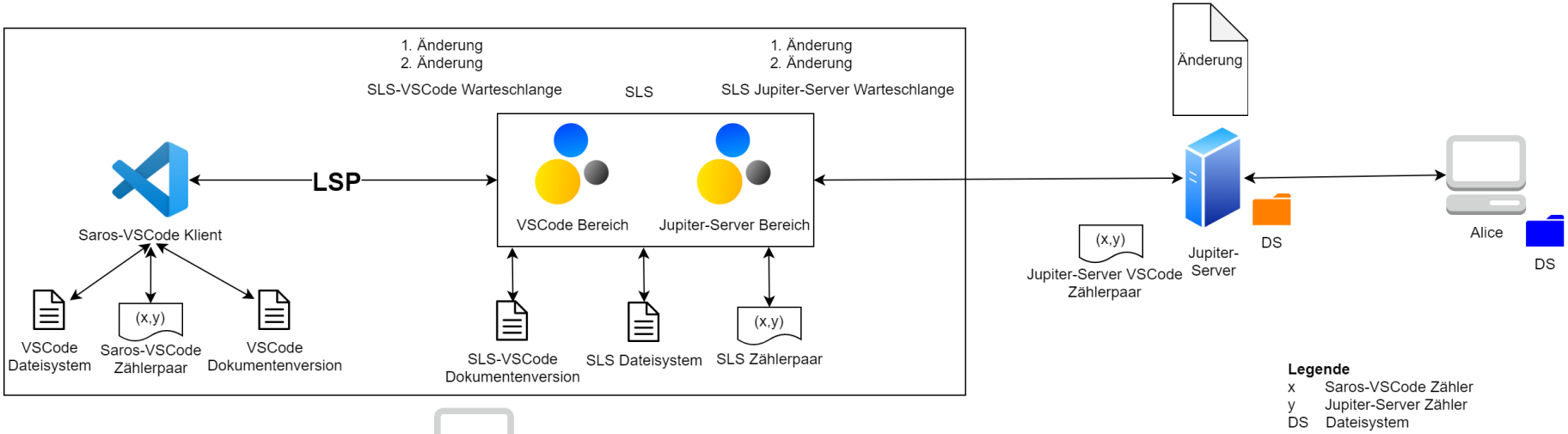


Freie Universität Berlin

Finale Lösung - Architektur



Finale Lösung – Beispiel



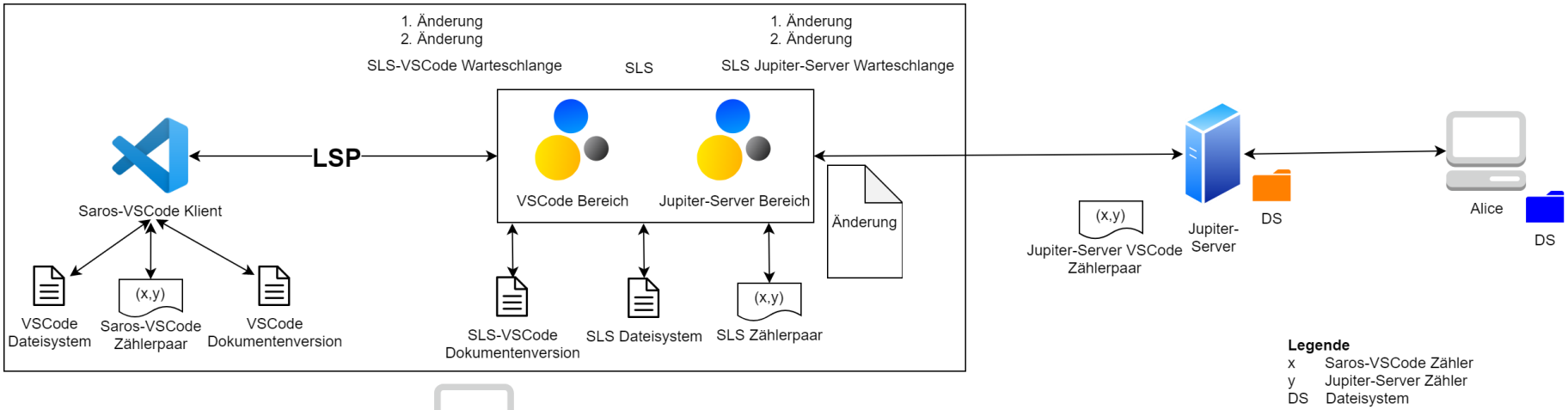
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,0)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,0)

Freie Universität Berlin

Finale Lösung – Beispiel



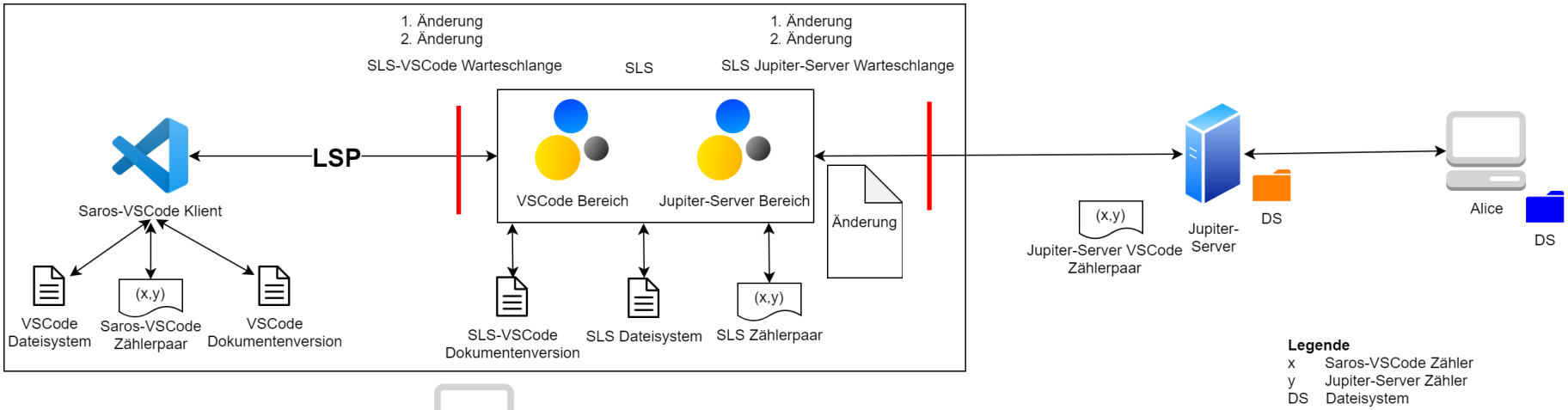
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,0)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



Bob

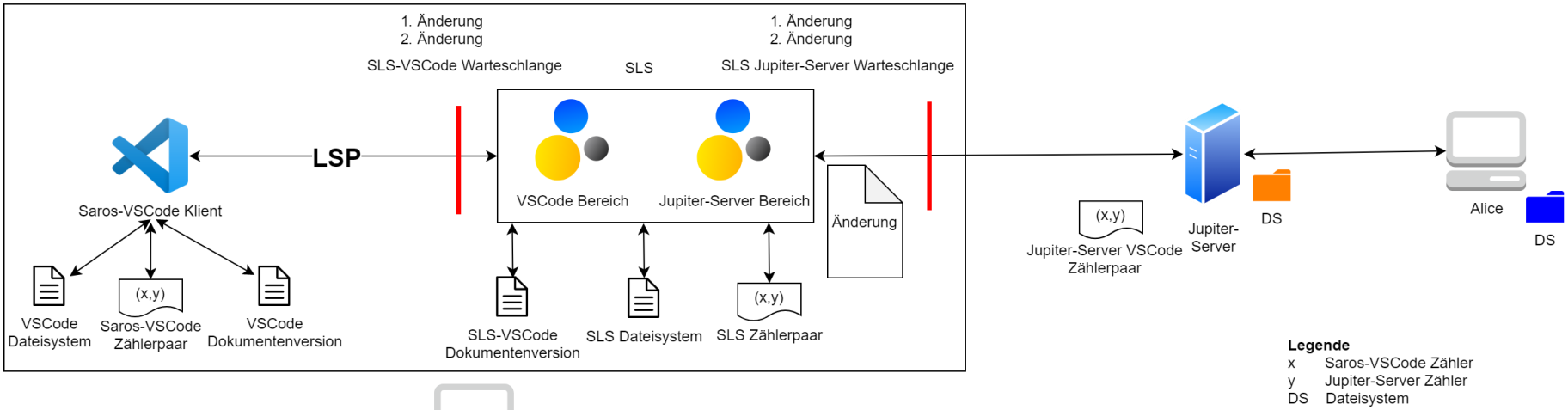
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,0)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



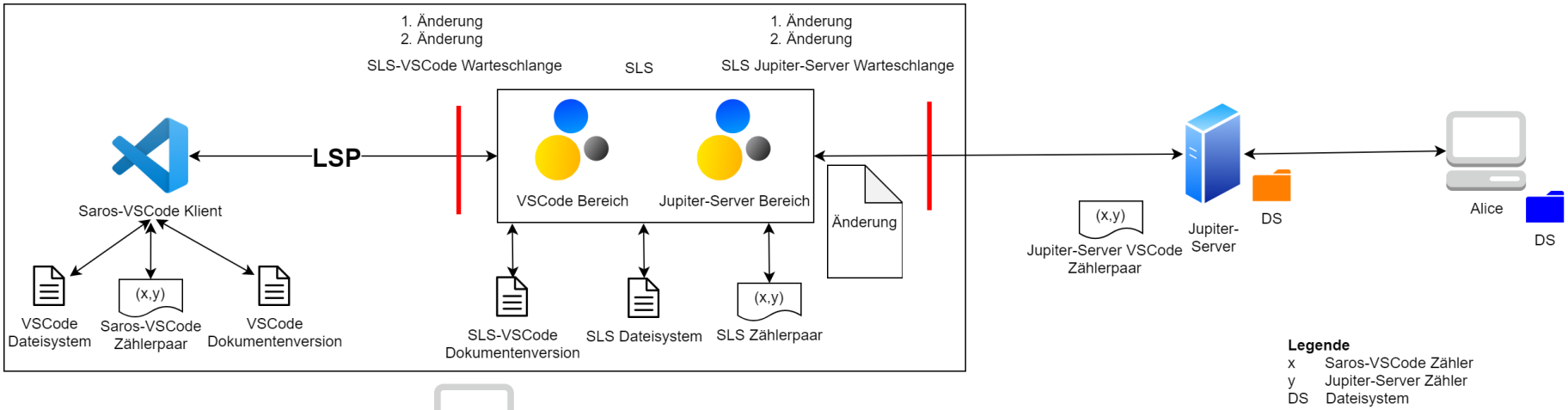
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,0)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



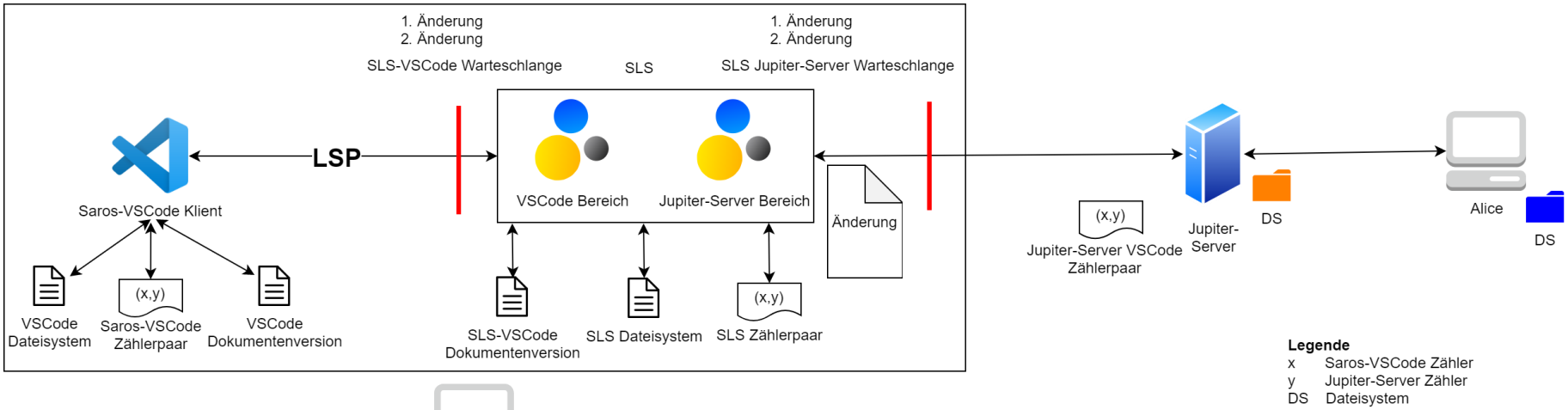
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,0)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

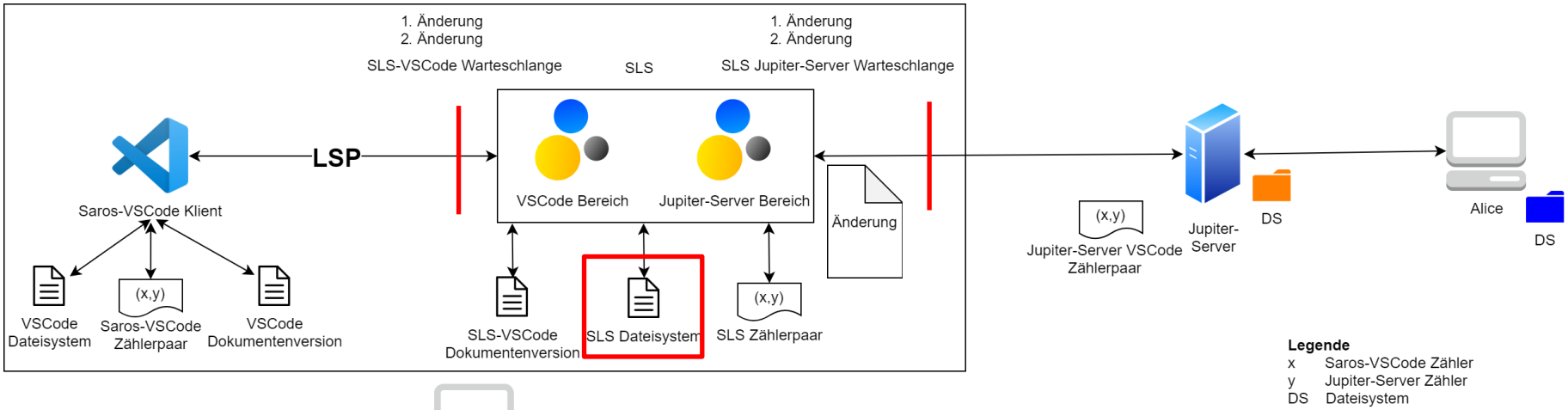
Bob

SLS Zählerpaar:
(0,0)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



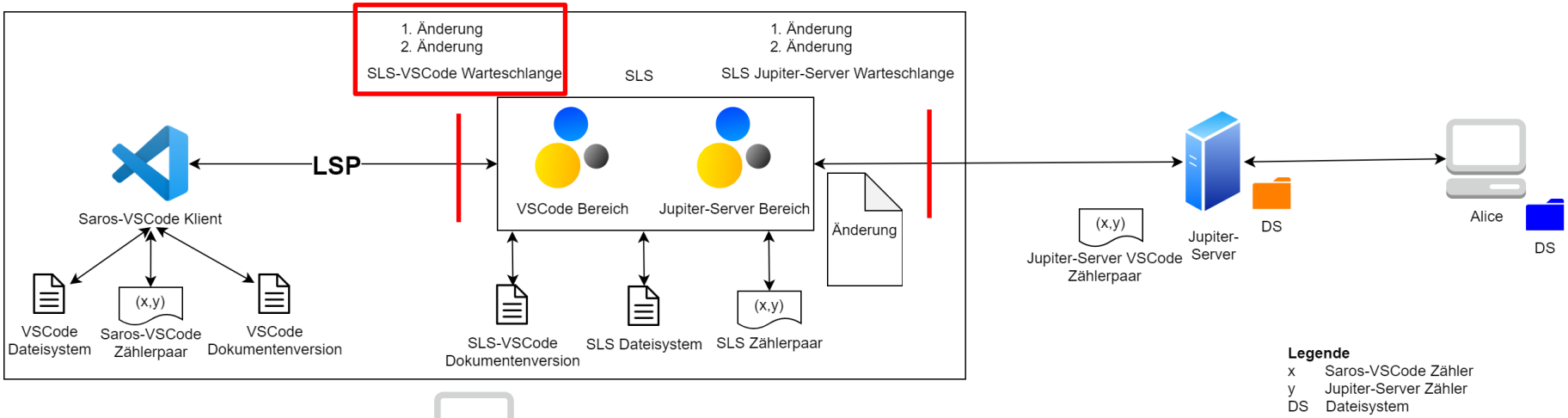
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



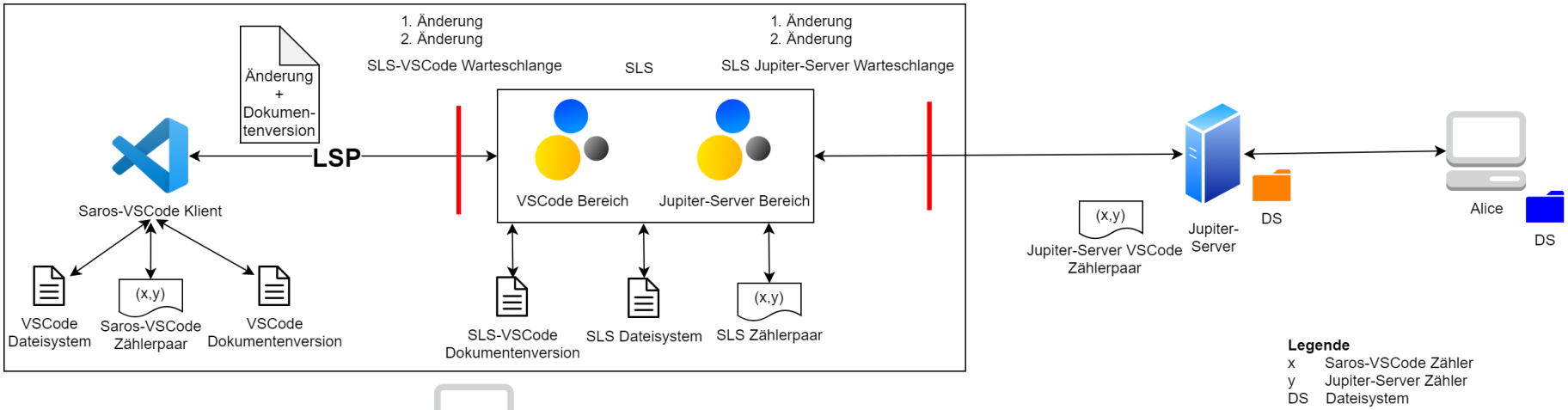
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



Bob

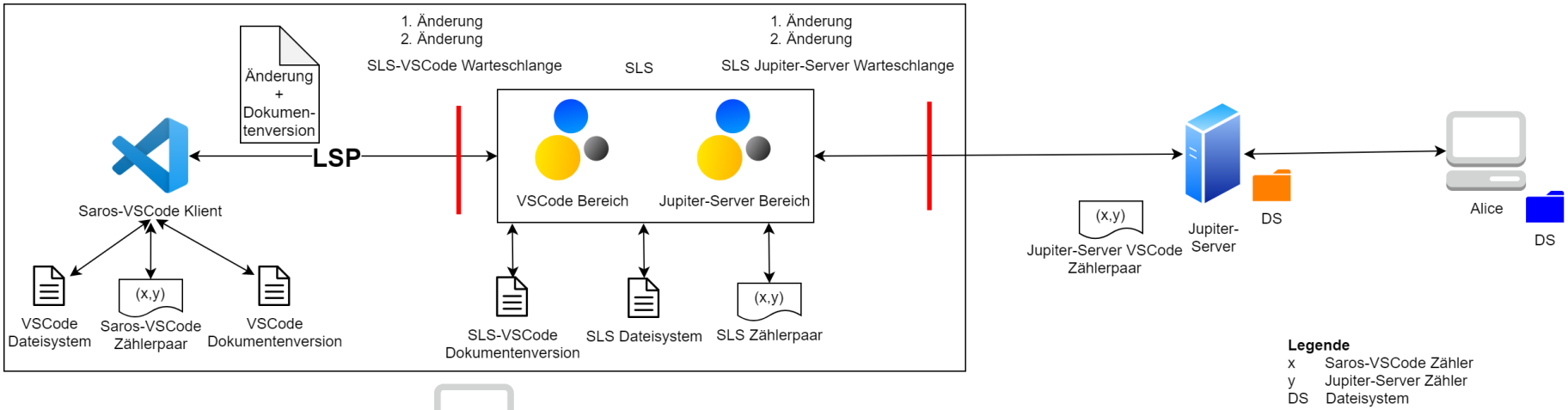
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



Bob

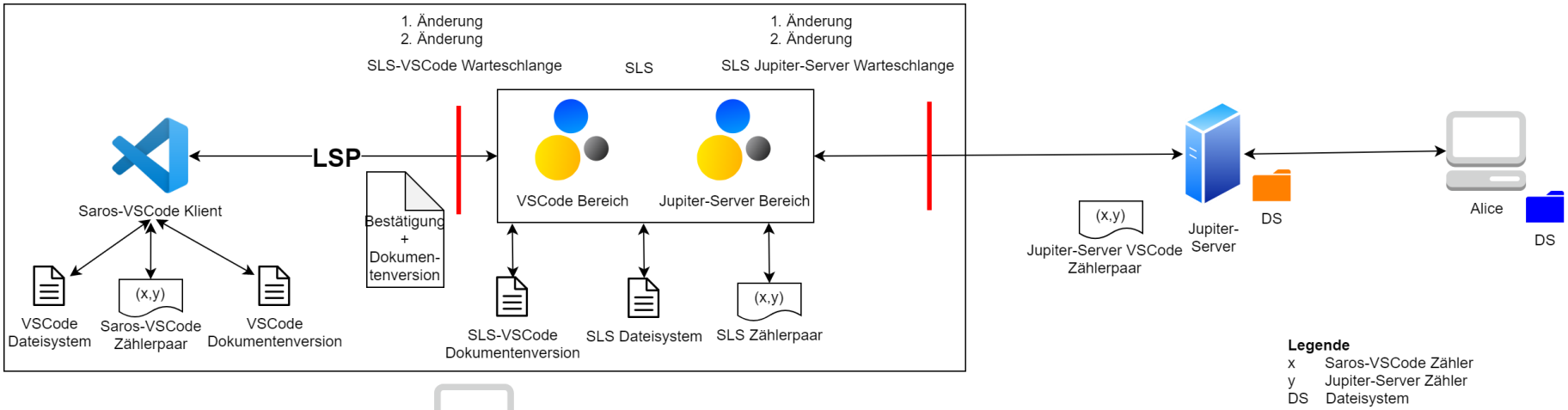
Saros-VSCode Zählerpaar:
(0,1)
VSCode Dokumentenversion:
1

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



Bob

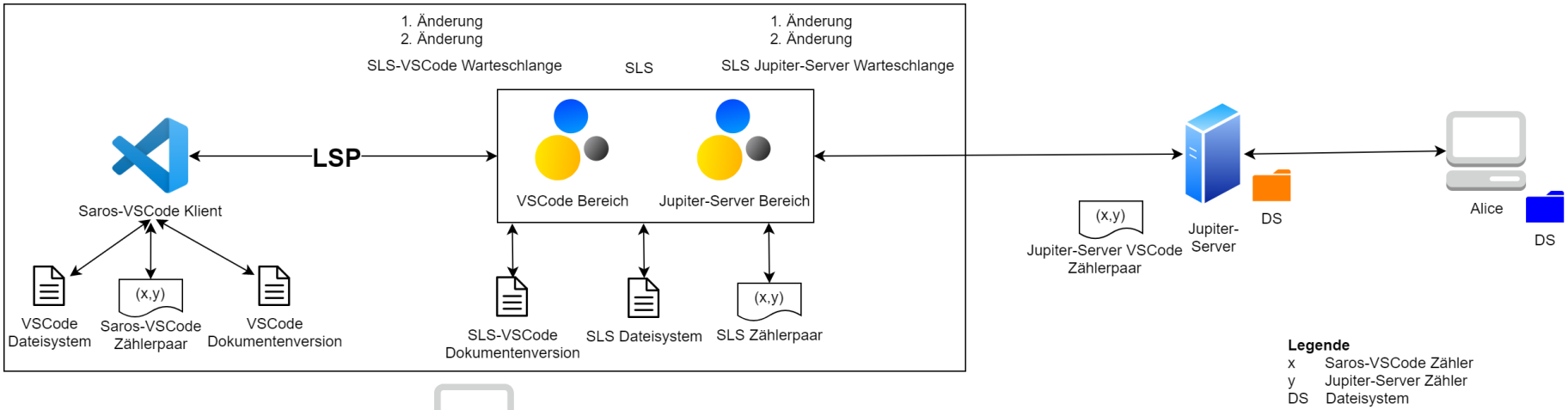
Saros-VSCode Zählerpaar:
(0,1)
VSCode Dokumentenversion:
1

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
1

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Beispiel



Bob

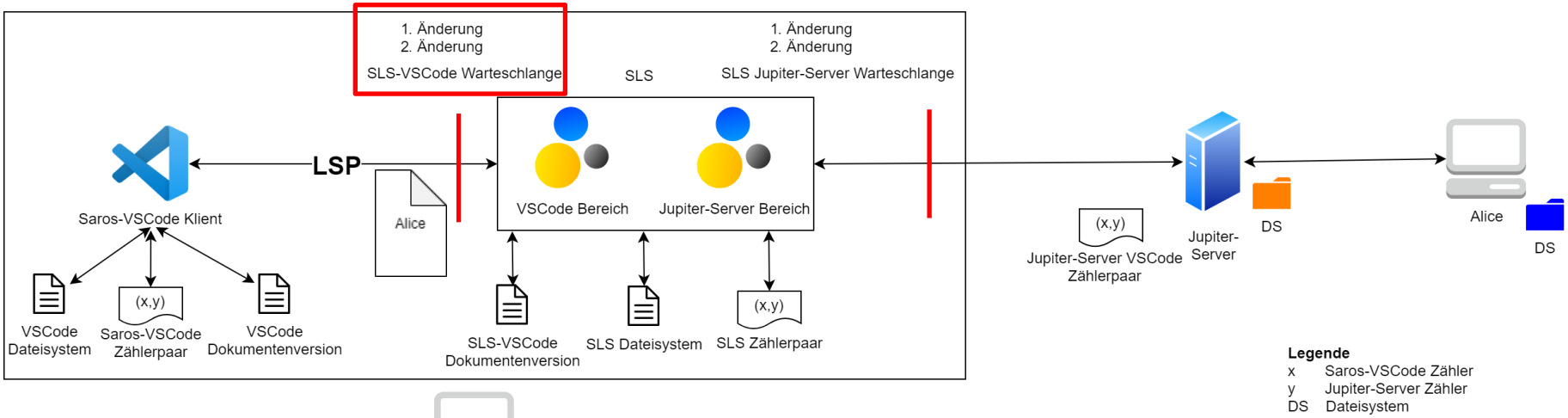
Saros-VSCode Zählerpaar:
(0,1)
VSCode Dokumentenversion:
1

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
1

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Problembeispiel



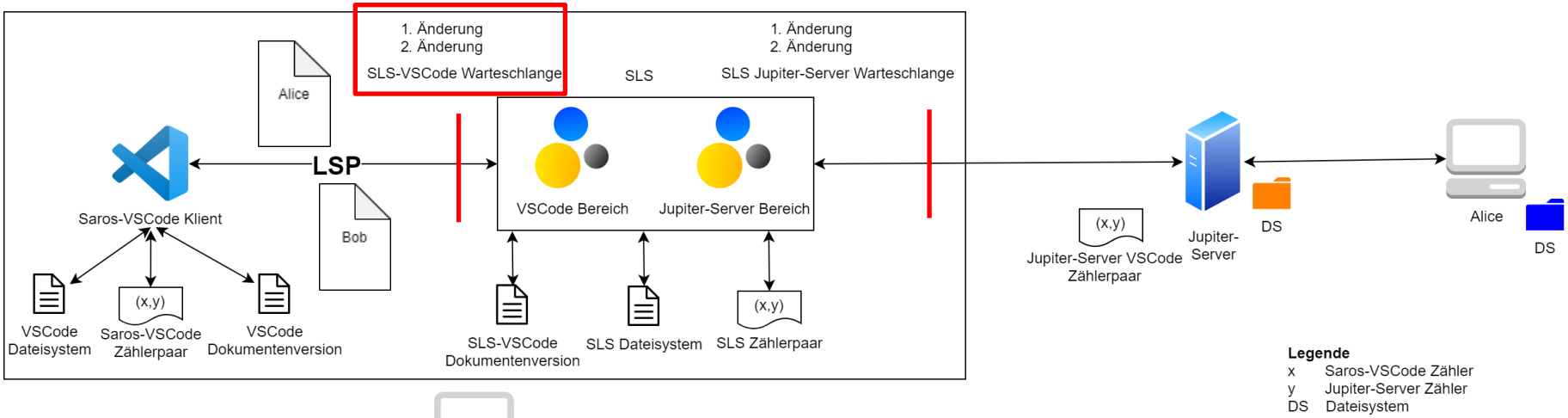
Saros-VSCode Zählerpaar:
(0,0)
VSCode Dokumentenversion:
0

Bob

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Finale Lösung – Problembeispiel



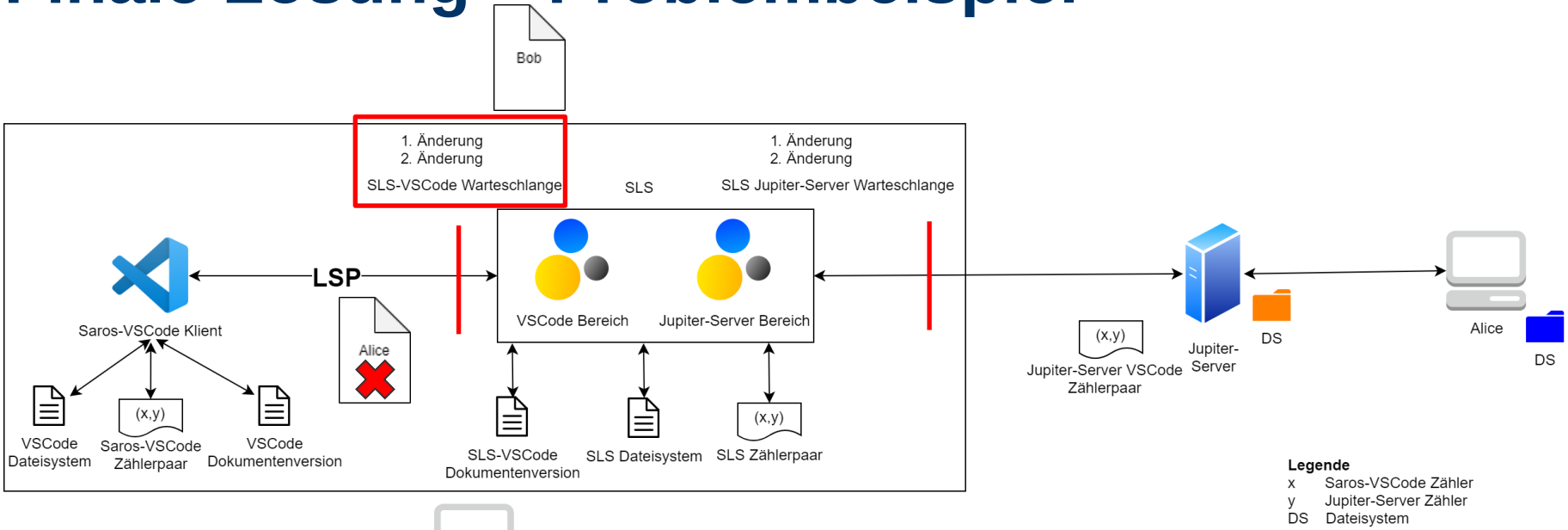
Saros-VSCode Zählerpaar:
(1,0)
VSCode Dokumentenversion:
1

Bob

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Finale Lösung – Problembeispiel



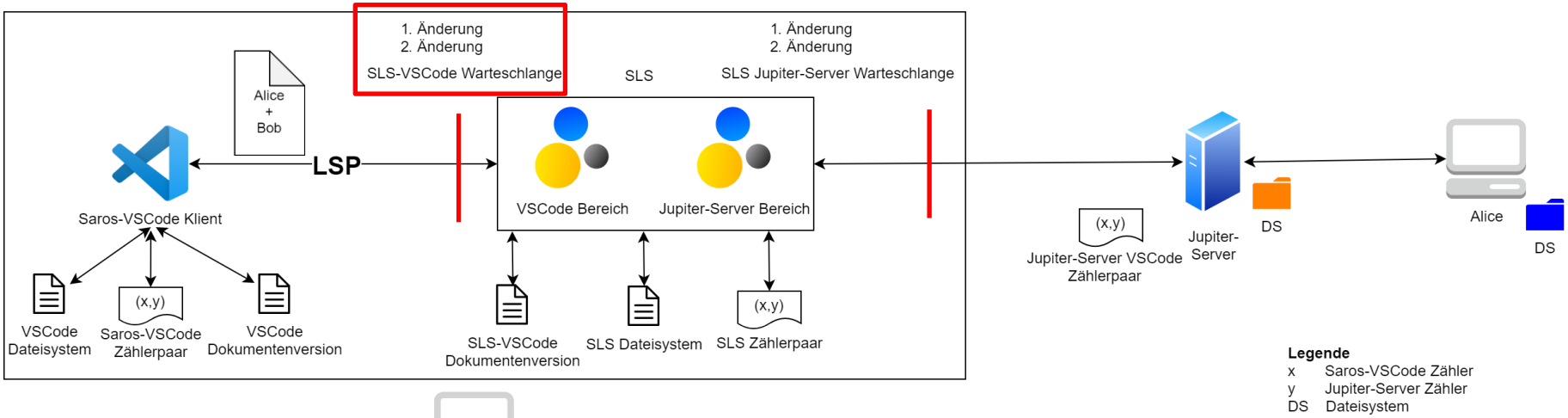
Saros-VSCode Zählerpaar:
(1,0)
VSCode Dokumentenversion:
1

Bob

SLS Zählerpaar:
(0,1)
SLS-VSCode Dokumentenversion:
0

Jupiter-Server VSCode Zählerpaar:
(0,1)

Finale Lösung – Problembeispiel



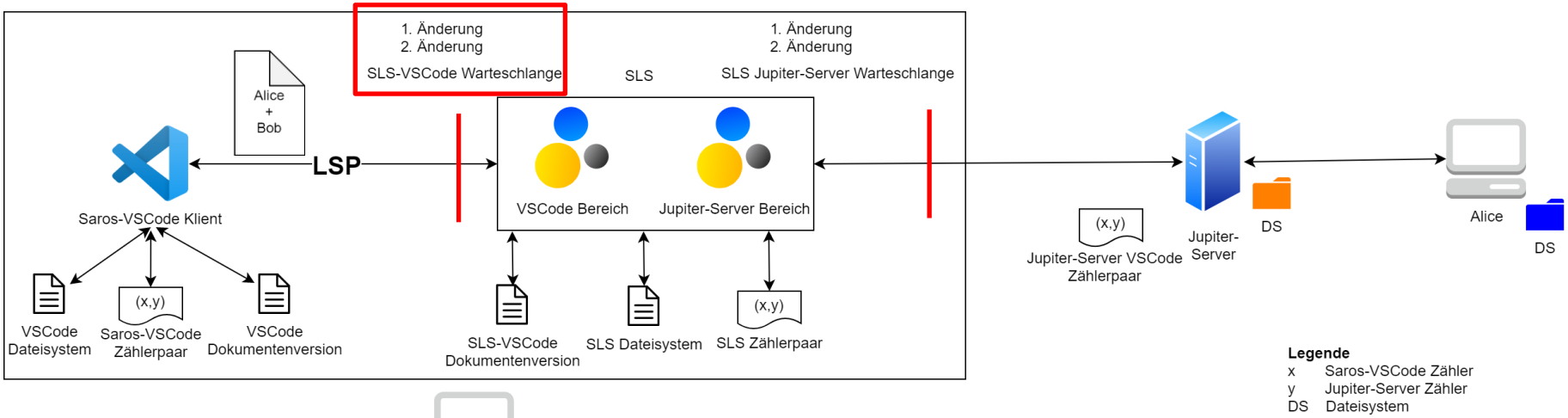
Saros-VSCode Zählerpaar:
(1,0)
VSCode Dokumentenversion:
1

SLS Zählerpaar:
(1,1)
SLS-VSCode Dokumentenversion:
1

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Problembeispiel



Bob

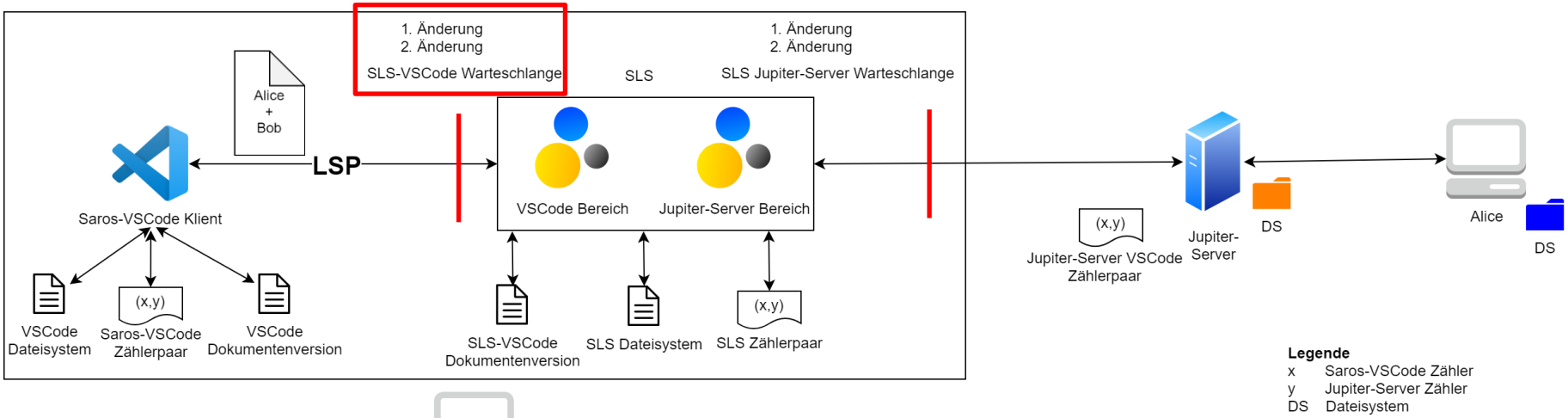
Saros-VSCode Zählerpaar:
(1,1)
VSCode Dokumentenversion:
2

SLS Zählerpaar:
(1,1)
SLS-VSCode Dokumentenversion:
2

Jupiter-Server VSCode Zählerpaar:
(0,1)

Freie Universität Berlin

Finale Lösung – Problembeispiel



Bob

Saros-VSCode Zählerpaar:
(1,1)
VSCode Dokumentenversion:
2

SLS Zählerpaar:
(1,1)
SLS-VSCode Dokumentenversion:
2

Jupiter-Server VSCode Zählerpaar:
(1,1)

Fazit

- Erfolgreiche Lösung des Problemfalls

- Das finale Konzept erfüllt dabei alle Anforderungen

Ausblick

- Implementierung des Prototyps

- Anbindung weiterer Editoren über LSP an Saros und den SLS

Freie Universität Berlin

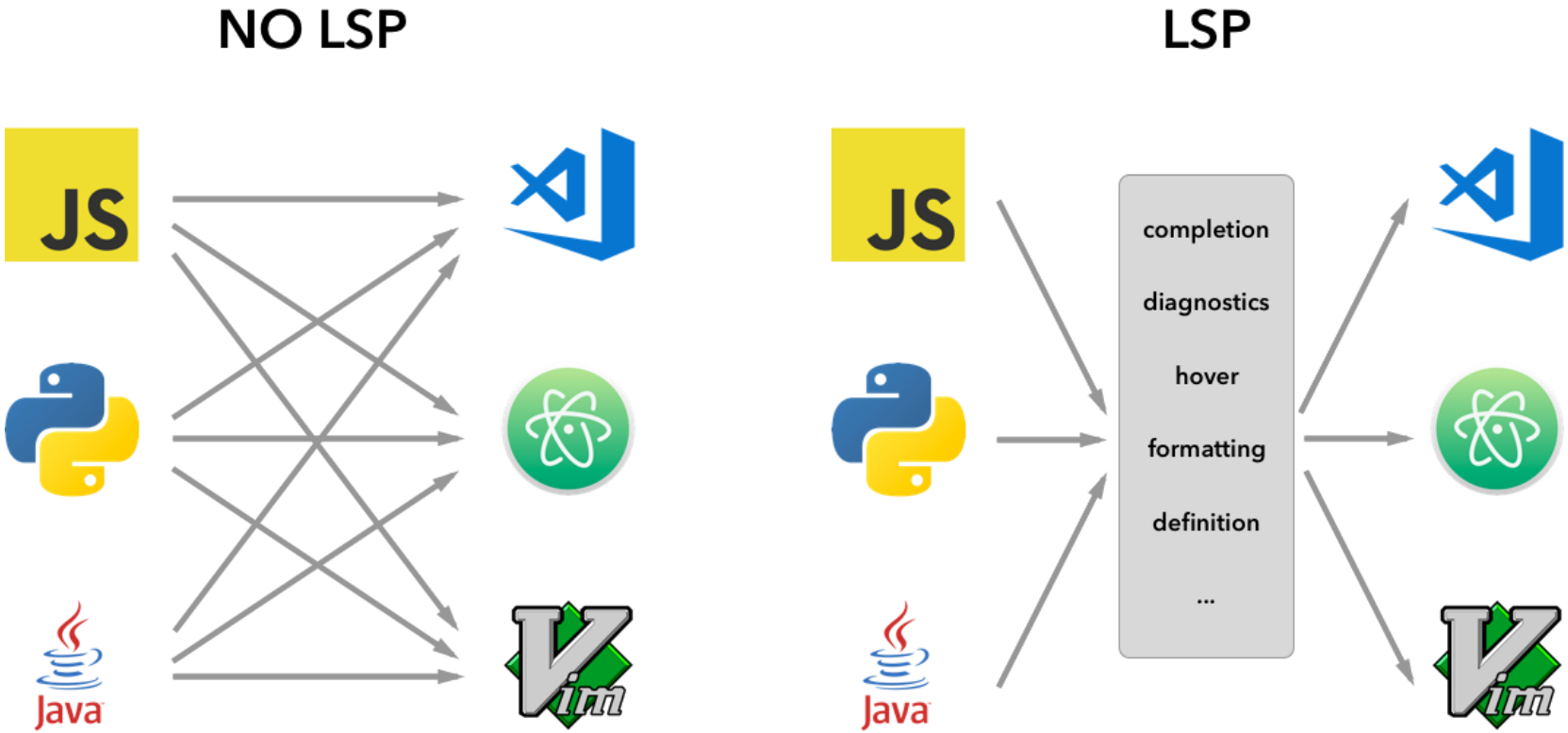
FRAGEN?

VIELEN DANK FÜR IHRE AUFMERKSAMKEIT

Freie Universität Berlin

BACKUP

Language Server Protocol



Quelle: https://code.visualstudio.com/api/language-extensions/language-server-extension-guide - 10.02.2021

Regeln zur Konfliktlösung des Jupiter-Servers

Wenn zum Beispiel Änderung c und s1 beide einen Text ersetzen, wird durch Jupiter ein finaler Stand erzeugt, welcher die Regionen aus beiden Änderungen entfernt und anschließend, sortiert nach der Startposition des ersten Zeichens des jeweils entfernten Textes, beide neuen Texte einfügt.

Regeln zur Konfliktlösung des Jupiter-Servers

Wenn ein Jupiter-Klient und Jupiter-Server Textinhalte an derselben Stelle einfügen möchten, wird immer der Text des Servers bevorzugt und ausschließlich eingefügt.

Regeln zur Konfliktlösung des Jupiter-Servers

*Ein anderer Fall ist, wenn zum Beispiel der Klient einen Text ersetzen möchte, welcher sich inmitten eines gelöschten Bereiches des Servers befindet. An dieser Stelle wird nach dem Löschen der neue Text immer eingefügt, um die Eingabe eines/einer Nutzer*in zu erhalten*

Regeln zur Konfliktlösung des Jupiter-Servers

1. Ein Set von Operationen sollte immer unter den Transformationsregeln geschlossen werden

2. Versuche, keine Nutzereingaben zu verwerfen

Warum das Saros-VSCode Zählerpaar und die Dokumentenversion?

- Dokumentenversion ist bestes Mittel zur Überprüfung der Datenstände

- Zählerpaare trotzdem für Anzahl der Änderungen in der Konfliktlösung wichtig

- Somit beide unerlässlich