

# Simple Multiuser Scenes for the 3D Web

Experiences with the Network Sensor / EventStreamSensor

By an anonymous member of the X3D Community

[x3d-public@web3d.org](mailto:x3d-public@web3d.org)

# Management Summary

- This Slide Set
  - Is based on experience with X3D v3.3, based on BS Contact and BS Collaborate of Bitmanagement Ges.m.b.H. in a testing environment for an experimental application of X3D (SrrTrains v0.01)
  - Does NOT explain X3D v3.3 NOR does it explain the Network Sensor
  - Lists three known use cases of the Network Sensor Node
  - Suggests a fourth use case of the Network Sensor Node and explains the required extensions
  - Explains the problem of collision of stream names in a scene that is authored by several or many authors

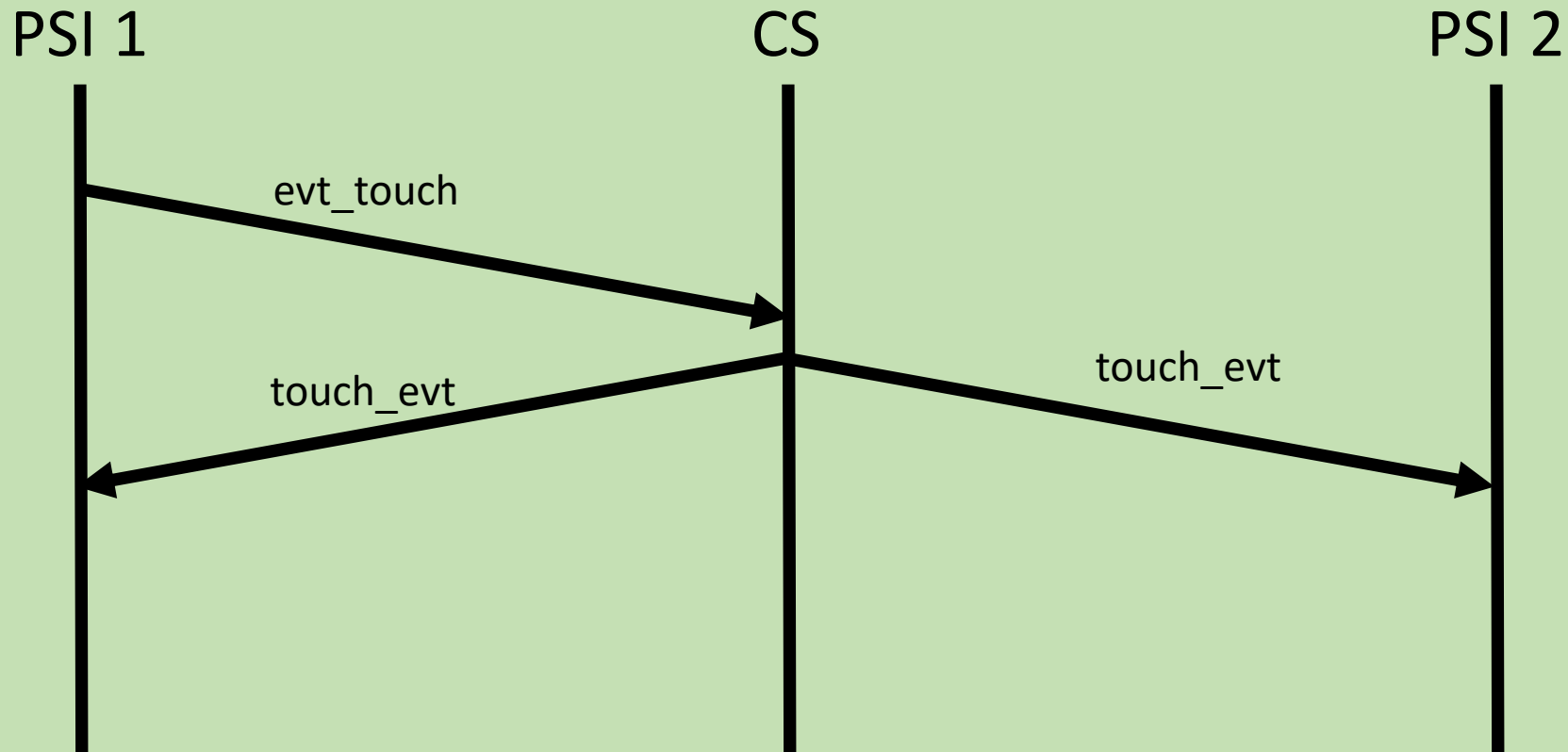
# Used Terms

- Web3D Browser – BS Contact (registered trade mark)
- Collaboration Server (CS) – BS Collaborate (registered trade mark)
- Personal Scene Instance (PSI) – one instance of the multiuser scene
- Simple Multiuser Scene (SMS) – a set of resources in the 3D Web that can be instantiated in a set of PSIs (multiuser session)

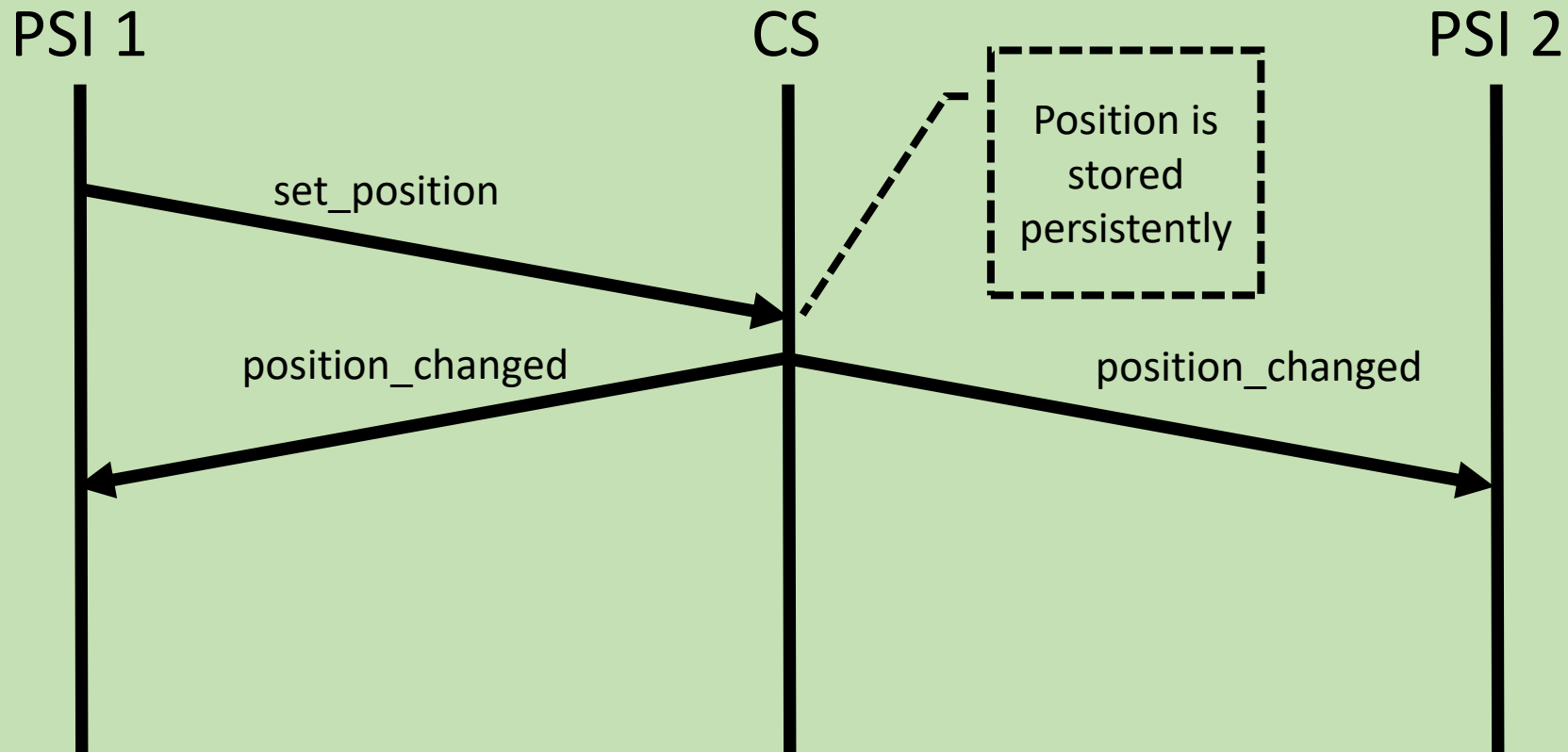
# Use Cases (1) – (4) of the Network Sensor

- (1) Event Distribution
- (2) (Re-)Setting States
- (3) Changing States by Predefined Server Side Calculations
- (4) (New) Changing States by Selfwritten Client Side Calculations

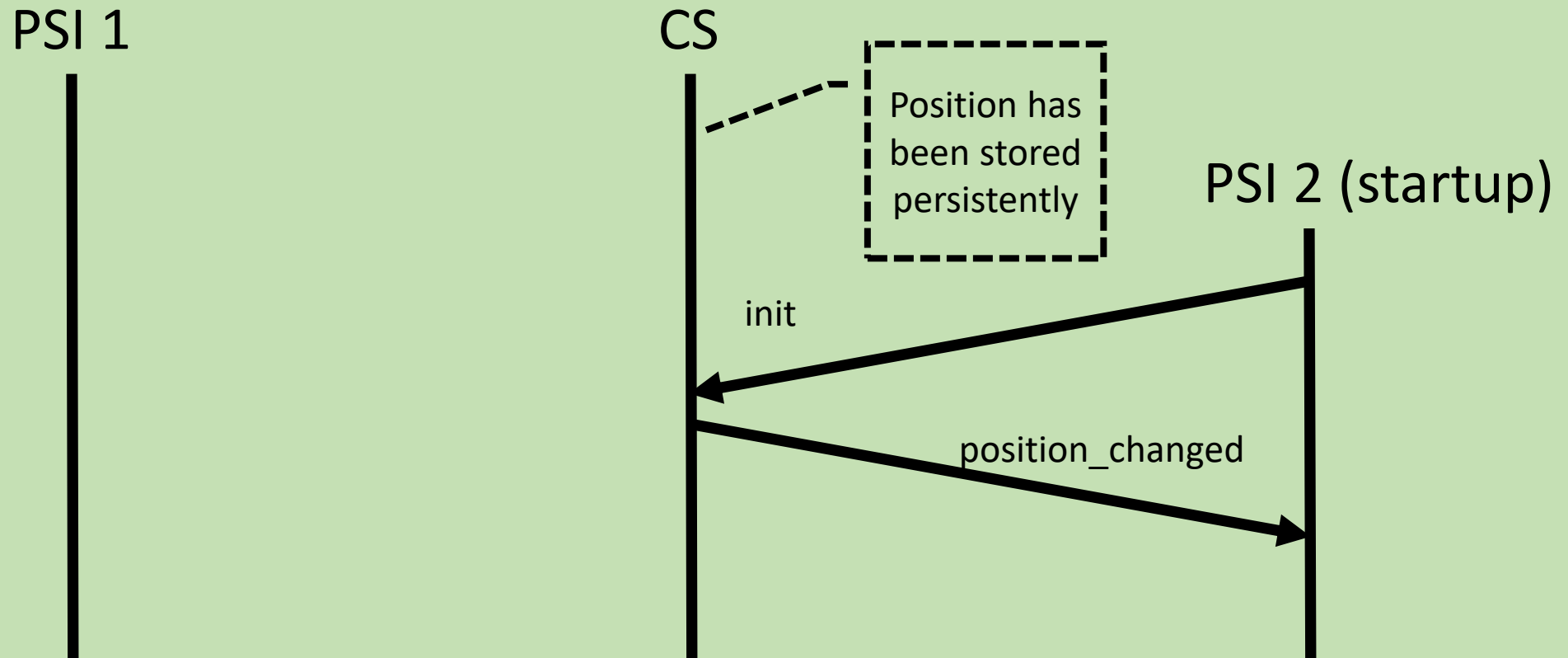
# Use Case (1) – Event Distribution (e.g. touch)



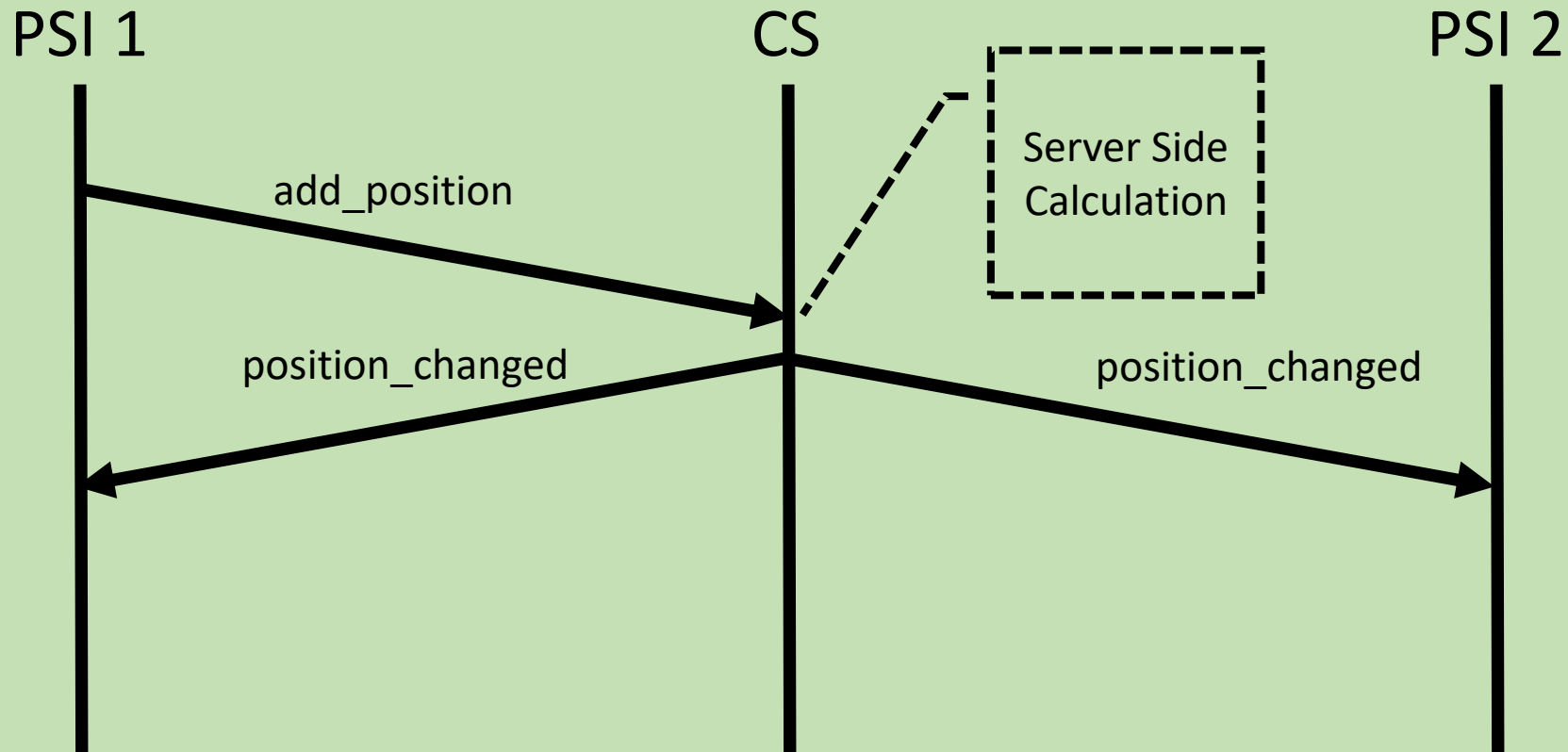
## Use Case (2) – Setting a State (e.g. position)



## Use Case (2) – Initializing a State



## Use Case (3) – Server Side Calculations (e.g. add)

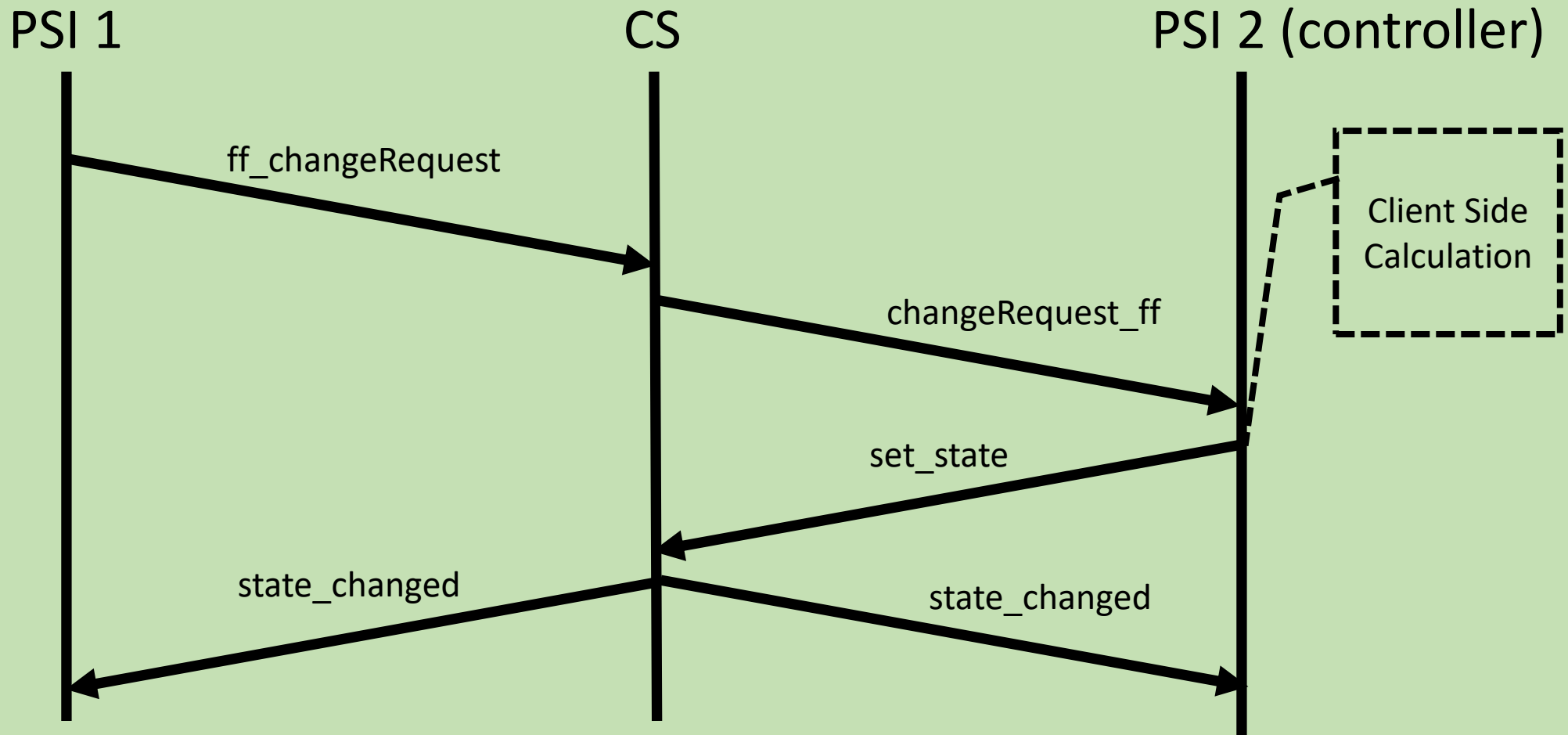




# Use Case (1) – (3) – Assessment

- Event Distribution is very simple
- Event Distribution can be used to start pre-defined animations
- Setting a state with „set\_“ implies a „REset“ of the state, because it does not take care about the current value of the state
- Changing a state with predefined server side calculations is better, because it cares about the current value of the state
- **Con**: server side calculations cannot be implemented/enhanced by the scene author unless he is the server operator
- **Solution of the Con**: Use Case (4)

## Use Case (4) – Client Side Calculations



# Use Case (4) – Assessment

- Client Side Calculations can be implemented „on top“ of X3D
- Very flexible and extensible
- It's rather complex – a controller role must be defined/maintained
- More network traffic
- Response times (approx. double time of Server Based Calculations)
- Paper with Required Extensions of the Network Sensor Node:

This paper is from January 2014, it contains broken links!!!

[https://areasharpa.files.wordpress.com/2018/04/smuos\\_03\\_sema\\_2018\\_04\\_27.pdf](https://areasharpa.files.wordpress.com/2018/04/smuos_03_sema_2018_04_27.pdf)

# Problem – Collision of Stream Names

- Assuming: A scene uses two models
- Model A from Author A
- Model B from Author B
- Model A contains a Network Sensor that uses streamName = „123“
- Model B contains a Network Sensor that uses streamName = „123“
- → The network traffic of both models cannot be separated at receive

# Solution – Meta Structure of the Scene

- Our Solution in Detail
  - Each scene consists of one and only one „frame“,
  - of one or more modules and of models within each module
  - The streamName of each network sensor must contain <moduleName> + <objId>
  - The frame sets the <moduleName> of each module
  - The module sets the <objId> of each model

# Additional Information

- Of related interest is a partial implementation at
  - 
  - X3D Example Archives: Basic, Networking, Network Sensor Connection Prototypes
  - <http://www.web3d.org/x3d/content/examples/Basic/Networking/NetworkSensorConnectionPrototypesIndex.html>
  -
- which includes a draft X3D Specification addition
  - 
  - <http://www.web3d.org/x3d/content/examples/Basic/Networking/NetworkSensorConnectionNodes.html>
  -
- Also please note summary of status on that directory page:
  - 
  - <http://www.web3d.org/x3d/content/examples/Basic/Networking/>
  -
- "Long-running efforts have attempted to define and build a new NetworkSensor node for X3D. Although useful design progress was made by the X3D working group, this work did not reach closure because author-written implementations did not appear to be possible using X3D prototypes encapsulating sandbox-restricted JavaScript network access from within an HTML browser. Further implementation and evaluation work might someday be pursued using an X3D browser implementation."