

Title	異なる性質のシミュレータを統合する都市のデジタルツイン基盤の研究
Author(s)	森田, 隆心
Citation	
Issue Date	2023-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/18335">http://hdl.handle.net/10119/18335</a>
Rights	
Description	Supervisor: 丹 康雄, 先端科学技術研究科, 修士 (情報科学)

修士論文

異なる性質のシミュレータを統合する都市のデジタルツイン基盤の研究

2110173 森田 隆心

主指導教員 丹 康雄  
審査委員主査 リム 勇仁

北陸先端科学技術大学院大学  
先端科学技術研究科  
(情報科学)

令和5年2月

## Abstract

Digital twin refers to a technology that senses information from the real world and reproduces and simulates it in a virtual space. It creates virtual models of urban buildings, spaces, infrastructure, people, climate, and all other information, senses the constantly changing information, and reproduces it on a computer. Furthermore, a key feature of this technology is that simulations are performed in the reproduced space to predict the future, which is then reflected again in the real world. Current digital twins are created for individual purposes and are used vertically. Each digital twin has an individual purpose, but the target city is the same, sharing space and time. If there is information shared in physical space, it can be integrated in virtual space as well, and by exchanging information with each other, digital twins can generate interaction. However, there is still no talk of integrating digital twins. There is talk of using sensing data in simulations in various fields, but there is no integration of digital twins that would create interaction between simulators, such as passing the output results of a simulation to another simulator. Therefore, we will consider methods of integrating digital twins and discuss rules that digital twin producers should follow in order to participate in the integration.

This study proposes an urban digital twin infrastructure that integrates simulators of different nature. One of the challenges of integrating a digital twin is that the time within the simulations running on the digital twin is different. Because of the different way time moves forward, there is a possibility of using data from a time that is not intended. In addition, the digital twin may recalculate according to the sensed data. Two types of data can be created: data generated in the past and data newly generated by recalculation. The ability for other digital twins that are working together to understand this is necessary.

Therefore, this study proposes the management of digital twins by "coordinators and messages" and data management by "shared database. Time stamps are attached to stored data to ensure that there are no inconsistencies in the causal relationships between the data used. In addition, messages will be used to notify when the digital twin starts and stops, and when values are updated by recalculation. When deleting data in the database, the digital twin asks the coordinator if there are any other digital twins using the data. If there is a digital twin using the data, it stops. The digital twin also makes a decision whether or not to use the values updated by the recalculation. The data not to be used are labeled, and if no digital twin using them exists, they are deleted.

In the experiment, we considered only input/output and dependent data for three digital twins and verified that they work with the proposed digital twin infrastructure. We were able to achieve periodic startup and shutdown. In addition,

by optimizing the data generated by the recalculation, we succeeded in reducing the data by 40 65% compared to when all the data was saved.

## 概要

デジタルツインとは、現実世界の情報をセンシングして、仮想空間で再現、シミュレーションする技術のことである。都市の建物や空間、インフラ、人、気候などのあらゆる情報を仮想モデル化し、常に変化し続ける情報をセンシングし、コンピュータ上で再現する。さらに、再現した空間でシミュレーションを行い将来予測を行い、現実世界に再び反映することが大きな特徴である。現在のデジタルツインは個々の目的のために作成されており、垂直的な利用がなされている。それぞれのデジタルツインは個々の目的を持っているが、対象としている都市は同じであり、空間や時間を共有している。物理空間で共有している情報があるのであれば、仮想空間上でも統合し、お互いに情報を交換することで、デジタルツイン同士が相互作用を生むことができる。しかし、デジタルツインを統合する話はまだない。センシングデータを様々な分野のシミュレーションで利用するという話はあるが、シミュレーションの出力結果を別のシミュレータに渡すといった、シミュレータ同士の相互作用を生み出すようなデジタルツインの統合は行われていない。そこでデジタルツインを統合する手法について考え、統合に参加するためにデジタルツイン製作者が従うべき規則について考察する。

本研究では異なる性質のシミュレータを統合する都市のデジタルツイン基盤を提案する。デジタルツインを統合する課題として、デジタルツイン上で動いているシミュレーション内の時間が異なっているという点である。時間の進み方が違うため、意図していない時間のデータを使用する可能性がある。また、デジタルツインはセンシングしたデータに応じて再計算を行う場合がある。過去に生成したデータと再計算によって新たに生成されたデータの2種類ができる。これを連携している他のデジタルツインが把握する機能が必要である。

そこで、本研究では「コーディネータとメッセージ」によるデジタルツインの管理と「共有データベース」によるデータ管理を提案する。保存するデータにタイムスタンプをつけることによって使用するデータにの因果関係に矛盾がないようにする。また、メッセージによってデジタルツインの起動と停止、再計算による値の更新の通知を行う。デジタルツインはデータベース内のデータを削除する際、他にデータを使用しているデジタルツインがないかコーディネータに尋ねる。もし使用しているデジタルツインが存在すれば停止する。また、デジタルツインは再計算によって更新された値に対して使用するかどうか判断を行う。使用しないデータにはラベル付を行い、使用するデジタルツインが存在しなければ削除する。

実験では3つのデジタルツインについて入出力と依存データのみを考え、提案したデジタルツイン基盤で動作するか検証を行った。周期的な起動と停止を実現することができた。また、再計算によって生じたデータの最適化を行ったことにより、全てのデータを保存した時に比べて40～65%のデータを削減することに成功した。

# 目次

<b>第1章</b>	<b>はじめに</b>	<b>1</b>
1.1	研究背景	1
1.2	研究目的	2
1.3	本論文の構成	2
<b>第2章</b>	<b>デジタルツインの現状</b>	<b>3</b>
2.1	デジタルツインおよび関連技術	3
2.1.1	デジタルツインとは	3
2.1.2	IoT	7
2.1.3	センシング, アクチュエータ	8
2.1.4	データ	9
2.1.5	シミュレーション	11
2.2	連携技術	12
<b>第3章</b>	<b>デジタルツイン統合と課題</b>	<b>13</b>
3.1	デジタルツイン統合	13
3.2	デジタルツインシミュレータの種類	14
3.3	デジタルツイン統合の課題	16
3.3.1	デジタルツインに流れる時間	16
3.3.2	データの依存関係	17
3.3.3	再計算	20
3.4	一般的な連携方法	23
<b>第4章</b>	<b>デジタルツイン統合の提案手法</b>	<b>25</b>
4.1	提案する連携方法	25
4.1.1	長時間の差が出る場合	25
4.1.2	短時間の差が出る場合	27
4.1.3	周期的に起動する場合	29
4.1.4	再計算のデータ管理	29
4.2	デジタルツイン開発者の基準	30

<b>第5章 システム提案</b>	<b>31</b>
5.1 システム概要	31
5.2 メッセージ交換	32
5.2.1 シミュレーション完了時のメッセージ	34
5.2.2 DB内の過去データを削除するときのメッセージ	34
5.2.3 デジタルツインを停止させる時のメッセージ	35
5.2.4 停止しているデジタルツインを起動させるときのメッセージ	35
5.2.5 デジタルツインが再計算を行った時のメッセージ	35
5.3 データ保存	36
5.4 起動管理	37
5.4.1 停止と起動	37
5.4.2 周期的な起動	38
<b>第6章 実験</b>	<b>39</b>
6.1 実験対象	39
6.2 実験概要	41
6.2.1 構築したシステム全体図	41
6.2.2 実験環境	41
6.2.3 デジタルツインの設定と連携方法	42
6.3 実験内容	43
6.4 実験結果	44
6.4.1 実験1: 単体起動	44
6.4.2 実験2: 短期的な時間差と周期起動するデジタルツインの連携起動	45
6.4.3 実験3: 長期的な時間差と周期起動するデジタルツインの連携起動	47
<b>第7章 評価</b>	<b>50</b>
7.1 実験に対する評価	50
7.1.1 データリソースについての評価	50
7.1.2 連携についての評価	51
7.2 課題に対する評価	51
7.3 議論	52
<b>第8章 終わりに</b>	<b>53</b>

# 目次

1.1	都市の複数のデジタルツイン	1
1.2	デジタルツインの連携	2
2.1	デジタルツインのコンポーネント	4
2.2	IoT デバイスの推移と予測	7
3.1	デジタルツインの統合	13
3.2	デジタルツイン間のデータ	14
3.3	デジタルツインの時間軸	16
3.4	2つのデジタルツインの時間の進み方	17
3.5	デジタルツイン間の依存関係1	18
3.6	デジタルツイン間の依存関係2	19
3.7	デジタルツイン間の依存関係3	19
3.8	デジタルツインの再計算	20
3.9	デジタルツイン連携時の再計算	21
3.10	再計算時のデータ生成	22
3.11	再計算が複数回行われた時のデータ生成	23
3.12	バリア同期	24
4.1	極端な時間のずれ	26
4.2	コーディネータによるデジタルツインの停止	27
4.3	短期的な時間の差によるデータの依存	28
4.4	センサデータとシミュレーションデータの保存	28
4.5	周期的な起動	29
4.6	再計算と使用するデータ	30
5.1	システム全体図	31
5.2	連携初期メッセージ	33
5.3	MongoDB を使ったデータベース構造	36
5.4	長期的な時間の差	37
6.1	公民館デジタルツインの例	39
6.2	実験で使ったシステム全体図	41
6.3	想定するデジタルツインの依存関係	43

6.4	公民館デジタルツインの単体起動 . . . . .	44
6.5	防災デジタルツインの単体起動 . . . . .	44
6.6	短期的な時間差で動く公民館デジタルツインのドキュメント数 . . .	45
6.7	再計算による削除 . . . . .	46
6.8	短期的な時間差で動く防災デジタルツインのドキュメント数 . . . .	46
6.9	都市計画デジタルツインの周期起動 . . . . .	47
6.10	長期的な時間差で動く公民館デジタルツインのドキュメント数 . . .	48
6.11	長期的な時間差で動く防災デジタルツインのドキュメント数 . . . .	48
6.12	都市計画デジタルツインの周期起動2 . . . . .	49

# 表 目 次

2.1	東京都デジタルツイン掲載データ . . . . .	6
2.2	センサの種類例 . . . . .	8
2.3	アクチュエータの動力源 . . . . .	9
2.4	デジタルツインで扱うデータ . . . . .	9
3.1	シミュレータの分類 . . . . .	16
4.1	データに付加する内容 . . . . .	27
5.1	MQTT にて交換するメッセージ . . . . .	32
6.1	公民館デジタルツインの入出力 . . . . .	40
6.2	防災デジタルツインの入出力 . . . . .	40
6.3	都市計画デジタルツインの入出力 . . . . .	40
6.4	実験環境 . . . . .	41
6.5	docker イメージ . . . . .	41
6.6	python とライブラリ . . . . .	42
6.7	センサと取得間隔 . . . . .	42
6.8	出力データ . . . . .	42
6.9	依存する入力データ . . . . .	42
6.10	短期的な時間差と周期起動するデジタルツインの設定 . . . . .	45
6.11	長期的な時間差と周期起動するデジタルツインの設定 . . . . .	47
7.1	実験 2 での再計算 . . . . .	50
7.2	実験 3 での再計算 . . . . .	50
7.3	削減したドキュメントの割合 . . . . .	50
7.4	本研究の連携とバリア同期 . . . . .	51

# 第1章 はじめに

本章では、本研究の背景、目的、本文の構成について述べる。

## 1.1 研究背景

デジタルツインとは、現実世界の情報をセンシングして、仮想空間で再現、シミュレーションする技術のことである。都市の建物や空間、インフラ、人、気候などのあらゆる情報を仮想モデル化し、常に変化し続ける情報をセンシングし、コンピュータ上で再現する。さらに、再現した空間でシミュレーションを行い将来予測を行い、現実世界に再び反映することが大きな特徴である。デジタルツインの活用分野は、工場現場の生産管理から始まり、近年では都市計画や防災の分野でも注目されている。現在の日本は少子高齢化や過疎化、自然災害など様々な問題を抱えており、一つの手段としてデジタルツインを使った課題解決が挙げられる。現在のデジタルツインやシミュレーションは個別の目的ごとに作成されており、垂直的な利用がされてきた。図 1.1 は3つのデジタルツインシステムが物理空間上の都市のデータをも

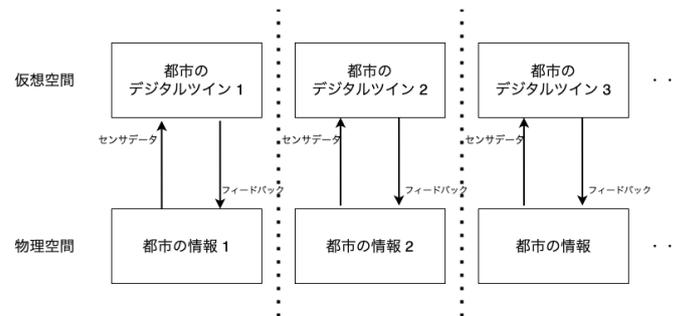


図 1.1: 都市の複数のデジタルツイン

と稼働している。これらのデジタルツインは一つの都市の中の事象を扱っているものであり、物理空間や時間を共有している。図 1.2 のように、これらを仮想空間で統合することで、デジタルツイン同士がお互いに協調し合うことができる。そこで、デジタルツインの独立性を担保しつつ、連携することも可能なフレームワークが求められる。

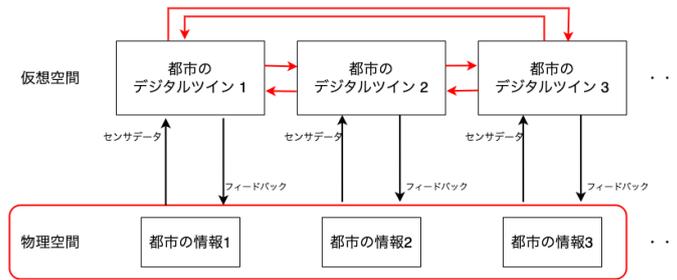


図 1.2: デジタルツインの連携

## 1.2 研究目的

本研究では、都市のデジタルツインにおける異なる性質のシミュレータを統合するシステムを提案する。都市を対象としたシミュレータは存在するが、それらを統合する話はない。シミュレータごとに処理時間や、対象としている時間など性質が異なる。しかし、同じ都市を対象としているためシミュレータ間には依存関係があると考えられる。そこで、シミュレータごとの性質を明らかにし、シミュレータの動作管理や交換するデータ管理について検討を行い、複数のシミュレータが連携をとるシステムを構築することを目的とする。

## 1.3 本論文の構成

- 第1章
  - 研究背景と研究目的について述べる。
- 第2章
  - デジタルツインについて調査した現状と連携技術について述べる。
- 第3章
  - デジタルツイン統合の課題について述べる。
- 第4章
  - 第3章の課題を解決する手法について述べる。
- 第5章
  - 本研究で提案するシステムについて述べる
- 第6章
  - 本研究で提案するシステムの実装について述べる。
- 第7章
  - 本論文のまとめと今後の課題について述べる。

## 第2章 デジタルツインの現状

本章ではデジタルツインとその関連技術として、以下の2項目について述べる。

1. デジタルツインおよび関連技術
2. 連携技術

### 2.1 デジタルツインおよび関連技術

#### 2.1.1 デジタルツインとは

デジタルツインとはIoT技術におけるセンサーデータとシミュレーション技術を組み合わせることで現実世界の対象物を仮想空間上に再現し、シミュレーションする技術とされている [1]。デジタルツインの起源は Michael Grieves と NASA の John Vickers が 2003 年に製品のライフサイクルに関する議論でこの概念が提示された。製品についての情報を手作業でかつ紙ベースで収集している時代に、Grieves と Vickers は製品の仮装モデルが製品ライフサイクルの管理の基礎を作ると考えた [2]。コンピュータ技術の進化により、製品の仮装モデルの構築と、シミュレーションが可能になった。そして、マイクロチップやセンサー、通知技術の進化は、物体の動作や環境などの情報を収集することを可能にした。シミュレーション技術の大幅な発展と製品からのデータの収集とフィードバックによって、物理製品をコンピュータ上に再現することができ、現在のデジタルツインの原型ができた [3]。NASA がデジタルツインについて最初に詳細に定義されて以降 [4]、デジタルツインは航空宇宙分野や製品ライフサイクルの分野での研究がされるようになる。最初のデジタルツインの定義では物理的な製品の情報を含む仮想表現であったが、Grieves[5]はこの定義を拡張し、物理的な製品とその仮想表現、物理製品と仮想にデータを双方向に接続する3つのコンポーネントで構成されるとした。(図 2.1)

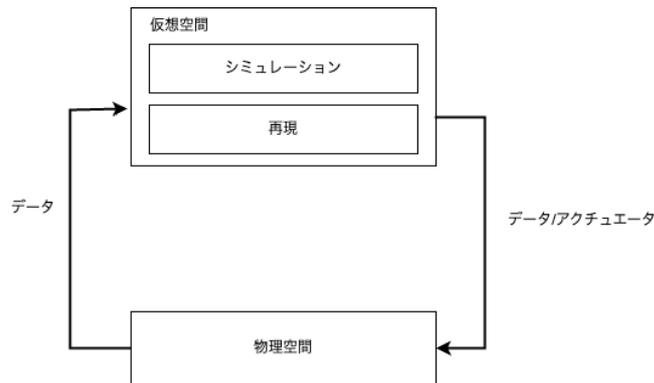


図 2.1: デジタルツインのコンポーネント

また、デジタルツインは物理製品の立体的なモデリングをすることであるという誤解が見受けられる。プロセッサの演算能力の向上によって物理世界の情報を三次元のグラフィックとしてコンピュータ上に再現することが可能になったが、デジタルツインの本質は、物理空間とその仮想表現と双方向通信の3つのコンポーネント図 2.1 と仮想空間上で行われるシミュレーションである。

2003 年以降、デジタルツインは関心が高まり、産学官が連携し研究、開発、および適用を試みている。IoT や解析技術、ビッグデータ、クラウドコンピューティング、エッジコンピューティング、セルラーネットワークなどの発展がデジタルツインの発展にも寄与している。デジタルツインが適用される分野はいくつか存在する。代表的な例をいくつか紹介する [6]。

### 製造

デジタルツインの多くは製造分野である。デジタルツインを適用するメリットとして、コスト、リスク、設計時間の削減、アフターサービスの改善、効率、メンテナンスの判断、セキュリティ、安全性と信頼性、製造管理、イノベーションの促進などが挙げられる [2]。

### ヘルスケア

人間が装着できる IoT デバイスが安価になりつつあるため、接続性が向上している。将来的に期待されているデジタルツインとしては人間のデジタルツインであり、薬の効能をシミュレートするために使用する。また、外科手術の計画や施術の適用にも期待されている。

### 都市

IoT の発展により、モノとの接続性が急激に増加している。都市のデジタルツイン計画の例を以下に示す。

## Virtual Singapore

Virtual Singapore[7] は公共や民間, 企業, 研究による使用を目的とした 3D デジタルプラットフォームである. 公共機関からのデータや IoT によるリアルタイムな動的データなどを統合している. これによりサービス提供のためのテストベッドとして利用することができる. 人流や周囲温度, 1 日の日照時間などは将来的な建築計画や都市計画の役に立つ.

## 東京都デジタルツイン実現プロジェクト

日本でのデジタルツインプロジェクトとして, 東京都デジタルツイン実現プロジェクトが挙げられる. 東京都全体をデジタルツインとして, 「2030 年に向けて, さまざまな分野でリアルタイムデータを活用し, 意思決定, 政策立案等で活用される完全なデジタルツインの実現をかがげている.」 [8] 現在は 3D 都市ビューアーにてさまざまなデータを閲覧することができる.

表 2.1: 東京都デジタルツイン掲載データ

データ種類	内容
3D デジタルマップ	<ul style="list-style-type: none"> <li>・建物モデル</li> <li>・地下通路</li> <li>・交通基盤</li> </ul>
統計・国土数値情報	<ul style="list-style-type: none"> <li>・行政界, 河川</li> <li>・国税調査, 区市町村別人口総数, 人口増減率</li> </ul>
防災	<ul style="list-style-type: none"> <li>・河川監視カメラ (リアルタイム)</li> <li>・海面監視カメラ (リアルタイム)</li> <li>・高潮浸水想定区域図</li> <li>・避難所</li> <li>・地震時危険区域</li> <li>・災害拠点病院</li> <li>・消火栓, 防火水槽</li> <li>・液状化予測図</li> <li>・火山ハザードマップ</li> </ul>
まちづくり	<ul style="list-style-type: none"> <li>・都市計画決定情報 GIS データ</li> <li>・ICT 活用工事 3D 点群データ</li> <li>・公共施設</li> <li>・スマートポール (センサ)</li> <li>・固定資産税, 線路価</li> <li>・街路樹</li> </ul>
モビリティ	<ul style="list-style-type: none"> <li>・都営バスロケーション (リアルタイム)</li> <li>・都営バス路線・バス停</li> <li>・時間制限駐車区間</li> <li>・自転車シェアリング</li> </ul>
エネルギー	<ul style="list-style-type: none"> <li>・優良特定地球温暖化対策事業所</li> <li>・環境局特定テナント評価制度</li> </ul>
その他	<ul style="list-style-type: none"> <li>・避難経路, ・点群データ</li> </ul>

表 2.1 は東京都デジタルツインの掲載データである。東京都はデジタルツイン実現のロードマップとしてまず利用者と分野を拡大させ浸透させることを初期の目標としている。現在実現できていることは主に可視化であり、リアルタイムセンサによるデータ取得はあるものの、シミュレーションや現実世界へのフィードバックなどはまだ行えない。現在はデジタルツイン基盤の構築を行なっている段階であり、データ仕様やルールを整備している最中である。今後シミュレータの開発などが行われる予定となっている。

## 2.1.2 IoT

IoT(Internet of Things) はあらゆる「モノ」がネットワークを通じて接続され、人間の介入なしに相互に情報を交換またはデバイスを制御する仕組みを指す。「モノ」とはセンサやアクチュエータだけではなく家電製品や住宅、自動車など多岐に渡り、拡大し続けている。図 2.2[9] は IoT デバイスの推移と予測を表している。通

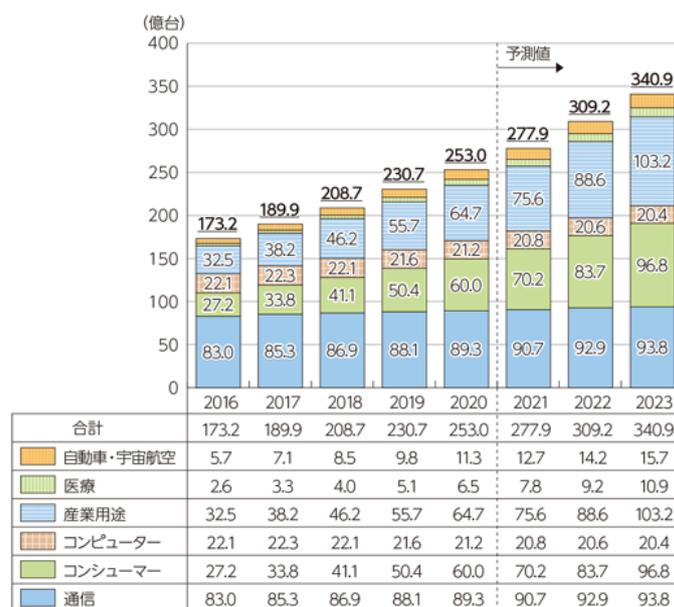


図 2.2: IoT デバイスの推移と予測

信分野については成熟しており、比較的成长率が低くなっている。ヘルスケアの分野、産業分野、スマートハウスや家電などのコンシューマ向けの分野の成長が今後見込まれている。都市のデジタルツインのセンシングやデータ収集には IoT の技術が必要不可欠である。

### 2.1.3 センシング, アクチュエータ

#### センサ

センサとは機械, 熱, 電磁気, 光, 化学, 生物などの性質の情報を機械が扱えるような信号に変換する装置のことを指す. デジタルツインを実現する上で, 物理空間上の情報を仮想空間でも扱えるように変換するのに役立つ. センサの分類と例を表 2.2[10] に示す.

表 2.2: センサの種類例

検出信号	測定値
機械量	圧力, トルク 流量, 体積, 質量, 幅, 傾き 速度, 加速度
熱	温度, 熱, 熱容量, 熱伝導率, エントロピー
電気	電流, 電圧, 電界, 電荷, 抵抗, 静電気容量, 誘電率, インダクタンス
磁気	磁界, 磁束, 透磁率, 磁気モーメント
光・放射線	照度, 波長, 位相, 赤外線, X線 CMOS
化学	pH, イオン, 濃度 水分, ガス, 酸素
生物	味, 臭気 タンパク質, グルコース, ホルモン, 酵素, 微生物
その他	GPS

表 2.2 に加えて近年ではカメラもセンサとして重要な役割を果たしている. 画像処理技術や機械学習の発展によって, 画像データから物体や文字などが認識できるようになった. カメラによる画像認識はセンサによる物理量の測定よりも汎用性が高い. 街に設置されたカメラにより, 人流や河川水位, 降雪量などを測定することができる.

デジタルツインや IoT を実現する上でこれらのセンサを単体で使用せず, センサネットワークを構築する. センサネットワークとは多数のセンサを各所に設置し, それらを協調させながら動作させ, 制御や観測を行うネットワークのことである. センサノードはセンサ, 無線通信チップ, マクロプロセッサ, 電源などで構成されており, センサノードの無線規格として, Bluetooth, ZigBee, Wi-Fi, NFC などがある.

#### アクチュエータ [11]

アクチュエータは入力されたエネルギーを運動に帰る装置のことである。アクチュエータは直線運動、回転運動、振動運動などを再現することができる。また、動力源は主に表 2.3 の 3 種類が存在する。

表 2.3: アクチュエータの動力源

動力源	例
電力	AC/DC モータ, ステッピングモータ, リニアモータ
油圧	油圧シリンダ, 油圧モータ
空気圧	空気圧シリンダ, 空気圧シリンダ

## 2.1.4 データ

デジタルツインを実現する上で、扱うデータについて述べる。データは主に、静的データと動的データに分類される。空間データは静的なデータである場合が多いが動的に変化する場合もある。これらのデータについてまとめたものを表 2.4 に示す。

表 2.4: デジタルツインで扱うデータ

データ分類	例
静的データ	<ul style="list-style-type: none"> <li>・統計データ</li> <li>・解析データ</li> <li>・文書データ</li> </ul>
動的データ	<ul style="list-style-type: none"> <li>・センシングデータ</li> <li>・ソーシャルデータ</li> </ul>
空間データ	<ul style="list-style-type: none"> <li>・GIS データ</li> <li>・POI</li> <li>・人(人流など)</li> <li>・標高モデル</li> <li>・衛星写真</li> <li>・3D 都市モデル (City GML)</li> <li>・BIM/CIM</li> <li>・点群</li> </ul>

### 統計データ

統計データとは集団現象を数量的に把握するために用いられる。性別や生年月日、

就業状態や住居の種類などを調べる国勢調査。住所、氏名、生年月日、性別、住民票コードなどが記載されている住民基本台帳。そのほか住宅土地統計調査、消費者物価指数などが挙げられる。政府機関が所有している統計データは e-Stat[12] などで公開されている。以下に都市計画に用いられる統計データの例を示す。

- 国勢調査

日本に住んでいるすべての人および世帯を対象とする国の最重要統計調査である。国内の人口や世帯を明らかにするために5年ごとに行われている。この統計情報は企業や研究などにも使用されている。

- 住民基本台帳

氏名や生年月日、住所などが記載されたものであり、市区町村の境界を超えて住所を移した移動者や、他の地域からの転入者などを知る事ができる。

- 人口予測データ [13]

国立社会保障・人口問題研究所が発表している人口の予測データである。コーホート要因法により、男女年齢別の生残率や出生率を算出し、2065年までの人口推移を予測している。

## 解析データ

あらかじめしていたシミュレーション結果や分析結果をさす。

## 文書データ

政府機関や企業などがだす電子報告書などを指す

## ソーシャルデータ

ソーシャルデータは SNS などで発信されている不特定多数のデータのことをさす。My City Report[14] は道路の劣化や不法投棄などの街の変化を市民が行政に向けて報告することができる。

## GIS データ

GIS(Geographic Information System) は地理情報を作成、管理、加工、可視化するシステムである。興味のある属性のみを表示させたり、異なる空間情報を重ね合わせて表示させて同時に可視化することができる。地形や道路、土地利用などの国土に関する空間情報を有する国土数値、境界線や測量の基準点などの情報を有する基盤地図情報などがある。

## POI

Point of Interest のことであり、避難所やトイレ、公共施設など特定の役割を果たす施設やランドマークのことを指す。

### 3D 都市モデル (City GML)

City GML は建物や道路, 川, 植生などのモデルを 3次元で表現するための国際標準である [15]. 通常の 3次元表現はジオメトリモデルであり, 形状以外の情報が含まれていなかった. CityGML は建物や道路などを区別して定義したセマンティックモデルであるため, 精密なシミュレーションをするのに役立つ. また, LOD(Level of Detail) を定義し, LOD が高くなると, 建物の内部構造まで表現することが可能になる.

日本では国土交通省が 3D 都市モデルのオープンデータである PLATEAU を公開し, 整備を進めている.

### BIM/CIM

BIM(Building Information Modeling) は建築物の設計, 建設, 運用にデジタルモデルを利用し, 設計に取り掛かる前にリスクを軽減し, 最適化を行うプロセスとされている [16]. BIM データは 3D 情報に加えて, 材料や, 素材, コストなどの属性を追加することができ, 建設前の設計の段階でコストやリスクについて議論することができる. 日本国内で, BIM は建築業で用いており, CIM(Construction Information Modeling) は土木業で活用するために用いられる. CIM は地形や地質などの自然環境を属性として追加することができる.

### 点群

レーザースキャナなどの 3次元測量によって得られた点の集合であり, 一つ一つの点は 3次元の位置情報を含んでいる. これらの点群を利用, 加工することによって 3次元モデルを生成することができる.

## 2.1.5 シミュレーション

シミュレーションとは対象となるモデルを作成し, 分析することである. シミュレーションモデルは, 微分方程式など連続的 (微小間隔) に変化する量をもとに対象系を記述する連続変化モデルと, 状態変化を引き起こす事象をもとに対象系を記述する離散変化モデルが存在する.

連続変化モデルは自然現象や, 製品の開発現場で用いられるような物理的な事象をシミュレーションする. 離散モデルは交通の渋滞や避難誘導, 待ち行列, 株式市場シミュレーションなどに用いられる.

都市の人間の動きをシミュレーションとしてマルチエージェントシミュレーション (MAS) が注目されている. MAS とは自律的に行動するオブジェクトをエージェントとし, エージェントの行動ルールとエージェント間の相互作用をモデルとして記述したものである. 複数のエージェントが個々のルールに従いながら同時に行動することによって相互作用を受けながらシミュレーションが実行される. これによ

り個々の動きだけでは予測できない複雑系の解析に用いられる [17].  
都市のシミュレーションとして防災や都市計画などが挙げられる.

### 防災シミュレーション

自然災害の多い日本では災害シミュレーションの研究が多く行われている. 津波のシミュレータとして国立研究開発法人海洋研究開発機構などが開発した JAGURS[18], 地震シミュレーションには国立研究開発法人防災科学技術研究所による GMS[19] などがある. いずれも微分方程式などを用いた連続変化モデルである. 一方, 災害シミュレーションには避難予測をするシミュレータがある. 非常事態時の人間行動を予測することは難しい. 身体的要因, 心理的要因, 社会的要因など様々な事象が影響を及ぼす. また, 緊急時に取得する情報によっても行動が変化する. このようなモデルをシミュレーションするには MAS などの離散シミュレーションが用いられる [20].

### 都市計画シミュレーション

都市計画シミュレータは, 人口動態や施設の存続などをシミュレーションするものであり, 都市の運営の手助けをする. 日本の地方の過疎化は大きな問題であり, 適切な予算配分や地域の活性化が求められている. My City Forecast[21] は国勢調査や国土数値, 住宅地図, 地方自治体の決算調査データなどを利用して, 2015 年から 2040 年までに想定される住居地域を可視化するものである. 主な指標として人口や年齢分布, 施設の存続, 行政コストなどをシミュレーションすることができる.

## 2.2 連携技術

シミュレーションを連携する技術として High Level Architecture(HLA) がある. HLA は米国防省の主導により標準化された分散シミュレーションのための規格である. ミドルウェアを通してデータ配信や時刻同期を行う. シミュレーション・エミュレーション連携基盤である Smithsonian は MQTT による Publish/Subscribe 方式のメッセージ交換により同期連携をおこなっている. ARIA[22] は Smithsonian を利用し, 水害発生時の被害をリアルタイムに取り込みつつ, 被害予測を行う. 被害予測に合わせて避難情報を発表するタイミングを変更することで, 被害量がどのくらい軽減できるか定量的に判断することができる.

## 第3章 デジタルツイン統合と課題

本章では、デジタルツイン統合の意義と課題について述べる。

### 3.1 デジタルツイン統合

本研究では都市のデジタルツインを統合する基盤の提案を行う。都市のデジタルツインは物理空間（現実世界）の建物や空間、人やインフラ、気候などあらゆる情報を含む。都市のデジタルツインは都市計画や交通、防災、スマートハウスなど様々である。現在のデジタルツインは個々の目的のために作成されており、垂直的な利用がなされている。それぞれのデジタルツインは個々の目的を持っ

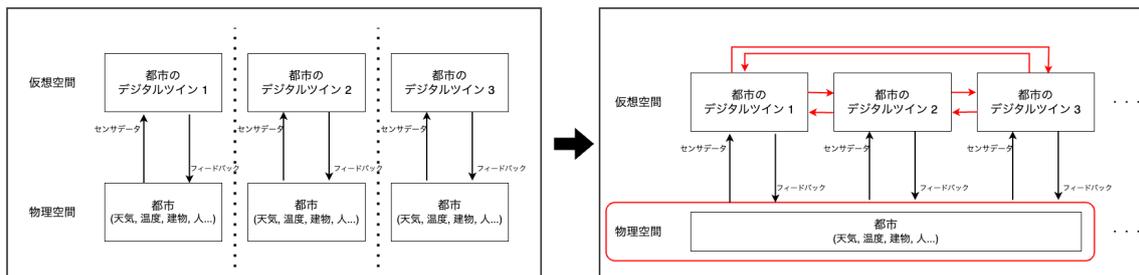


図 3.1: デジタルツインの統合

ているが、対象としている都市は同じであり、空間や時間を共有している。物理空間で共有している情報があるのであれば、仮想空間上でも統合し、お互いに情報を交換することで、デジタルツイン同士が相互作用を生むことができる (図 3.1)。しかし、デジタルツインを統合する話はまだない。センシングデータを様々な分野のシミュレーションで利用するという話はあるが、シミュレーションの出力結果を別のシミュレータに渡すといった、シミュレータ同士の相互作用を生み出すようなデジタルツインの統合は行われていない。図 3.2 は 3 つのデジタルツインがお互いに依存データを交換しながら実行されている様子を表している。通常のデジタルツインはセンサデータ、静的データ、動的データを入力としてシミュレーションを行う。静的データが他のデジタルツインによって動的に変化するのであれば、出力結果も変化する。この出力変化が別のデジタルツインにも影響を与える。

相互作用を生み出すことによって、単体のデジタルツインでは起こり得ない事態にも対応することができる。例えば、水害のデジタルツインによって対象都市が浸水

することが予想されているのにも関わらず、公民館などの公共施設のデジタルツインでイベントが行われているのは不自然である。デジタルツイン間で災害の情報が共有されているとしたら、公民館のデジタルツインはそれに対応できる。

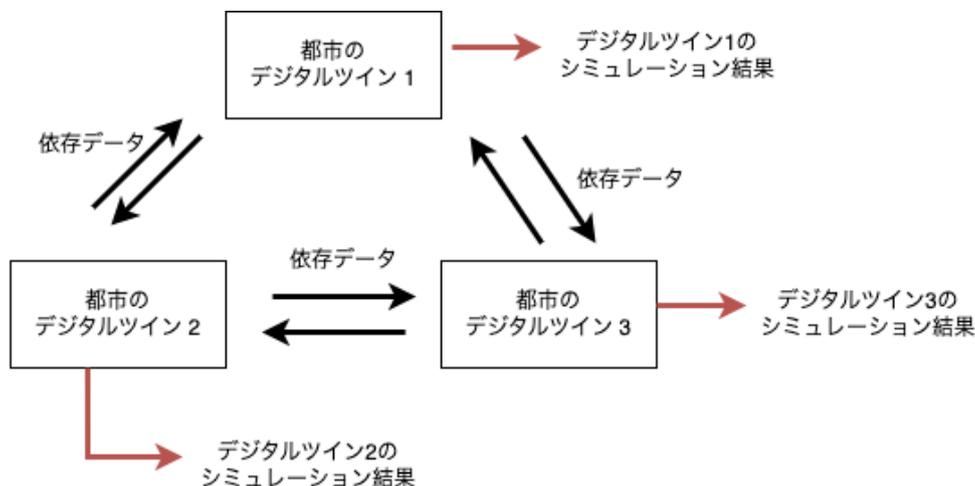


図 3.2: デジタルツイン間のデータ

デジタルツイン統合基盤は複数のデジタルツインを統合できるだけでなく、都市のデジタルツインであれば、その性質に関わらず追加や削除ができるのが好ましい。デジタルツイン製作者は、デジタルツイン統合基盤のルールに則り制作し、製作者同士がお互いのデジタルツインを意識しなくても連携できる仕組みを考える。連携するデジタルツインがお互いにわからなければ、相互作用が生まれえない可能性もあるが、それは相互作用を生まないデジタルツイン同士であっただけである。個別のデジタルツインは自身の目的のためのシミュレータを動かしており、デジタルツイン基盤を通して連携したとしても疎な関係となっている。このように個々のデジタルツインの独立性を担保しつつ、連携する事が可能なフレームワークが求められる。

## 3.2 デジタルツインシミュレータの種類

2.1.5で述べたようにデジタルツインのシミュレーションはいくつかの性質に分類することができる。そこでシミュレータの違いとして、A: 扱う時間の違い, B: 動作方法, C: モード変更, D: 処理時間の4つに分類した。

- 扱う時間  
扱う時間とはシミュレーションが対象としている時間軸のことを指す。

- 実時間  
実時間はリアルタイムシミュレータであり, 現実世界の時間とシミュレーション内の時間が同期されている.
  - 短期間  
短期間とは実時間の変化によって影響を受けるものである.
  - 長期間  
長期間とは実時間の変化にあまり影響を及ぼさないものである.
- 動作方法  
動作方法はシミュレータの起動方法である.
    - 連続した動作
    - イベントに応じた動作
- モード変更  
モード変更とは, 環境によってシミュレーション内容やアクセスするデータが変化するシミュレータのことを指す. 雪害を扱う防災シミュレータであれば, 通常の雪と豪雪では状況が変わる. 通常の雪害のシミュレータでは除雪車のルートを出力するものとなっているが, 豪雪時にはモード変更することによって, 避難計画や救助計画のシミュレータに変わる.
    - しない
    - する
- 処理時間  
処理時間によっても分類する事ができる. 実時間より遅いシミュレーションは解析に時間がかかるものである.
    - 実時間よりも速い
    - 実時間
    - 実時間よりも遅い

以上のシミュレータの分類についてまとめたものを表 3.1 に示す.

表 3.1: シミュレータの分類

扱う時間の違い	<ul style="list-style-type: none"> <li>・実時間 (リアルタイム)</li> <li>・短期間</li> <li>・長期間</li> </ul>
動作方法	<ul style="list-style-type: none"> <li>・連続した動作</li> <li>・イベントに応じた動作</li> </ul>
モード変更	<ul style="list-style-type: none"> <li>・しない&amp;・する</li> </ul>
処理時間	<ul style="list-style-type: none"> <li>・実時間よりも速い</li> <li>・実時間</li> <li>・実時間よりも遅い</li> </ul>

### 3.3 デジタルツイン統合の課題

デジタルツイン連携基盤の実現における課題を以下に示す.

#### 3.3.1 デジタルツインに流れる時間

デジタルツインの統合を考える上で, それぞれのデジタルツインの時間の進み方について考える.

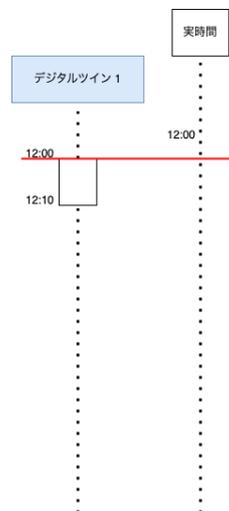


図 3.3: デジタルツインの時間軸

図 3.3 は, デジタルツイン 1 が 12:00 に起動して 12:10 までのシミュレーションをおこなったことを表している. デジタルツインの種類によっては解析に時間がかかり, 実時間よりもシミュレーション時間の方が長いものが存在するが, 多くは実時間よりも早くシミュレーションするデジタルツインである. また, デジタルツイン

はシミュレーションの時間と別に実時間の時計の機能もあり、センサの入力タイミングは実時間と同期している。

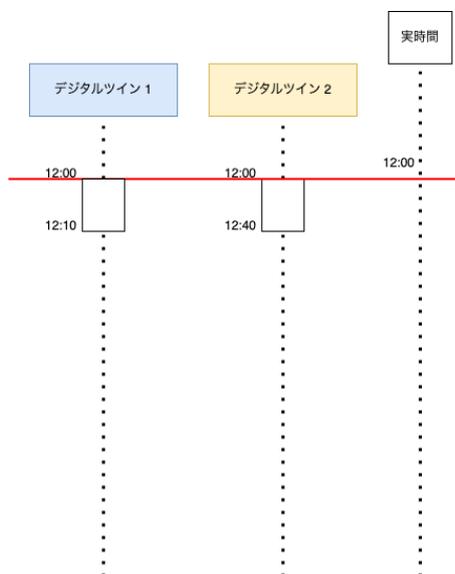


図 3.4: 2つのデジタルツインの時間の進み方

次に図 3.3に加えて、さらにデジタルツインを加えた場合の時間について考える。デジタルツイン 2 を追加した時の様子を図 3.4 に示す。デジタルツイン 1 は 12:10 までのシミュレーションをおこなっているのに対して、デジタルツイン 2 は同じ時間で 12:40 までのシミュレーションをおこなっている事がわかる。これはデジタルツイン 2 の方がシミュレータ内での時間の進み方が早いという事がわかる。

デジタルツイン 1 とデジタルツイン 2 を同時に起動して長時間動かし続けると両者の時間の進み方の違いにより、大幅な時間的ずれが生じてしまう。デジタルツインを統合する上で、デジタルツインの内部の時間の進み方の違いを考える必要がある。

上述でデジタルツインごとに進時間が違うことを説明した。このようにデジタルツインごとに異なる時間軸を持っている。

### 3.3.2 データの依存関係

前述した通り、デジタルツインはそれぞれ固有のシミュレーション時間(時間軸)を持っている。図 3.5 は 12:00 の時点で、デジタルツイン 1 は 12:10 までのシミュレーション、デジタルツイン 2 は 12:40 までのシミュレーションを完了している。次に 12:30 の地点でデジタルツイン 1 がセンサデータとデジタルツイン 2 のデータを利用してシミュレーションを行うとする(図 3.6)。すると、デジタルツイン 1 のシミュレータ内時間は 12:30 なのにも関わらず、デジタルツイン 2 が生成した 12:40

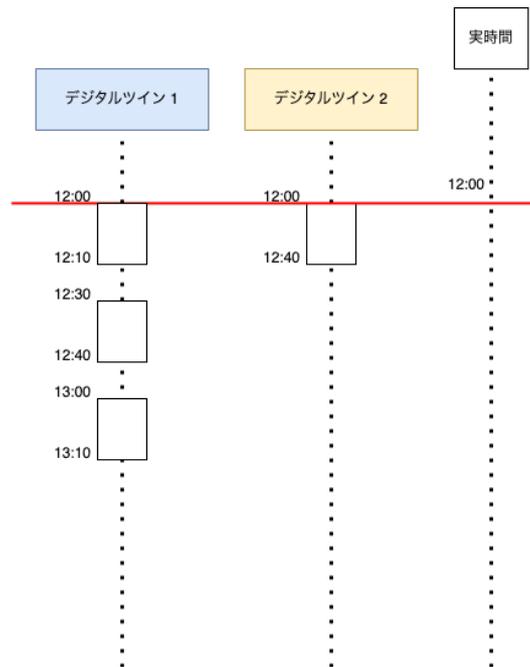


図 3.5: デジタルツイン間の依存関係 1

のデータを取り込んでしまう。つまり、データの因果関係を正しく維持できていない事がわかる。

図 3.6 のように未来のデータを使用するのは不自然である。本来であれば、デジタルツインがシミュレーションする時間より前の時間のデータを使用しなければいけない。図 3.7 のようにデジタルツイン 1 で 12:00 にシミュレーションを開始するならばそれ以前のデータ 12:30 にシミュレーションを開始するならば 12:30 より前に生成されたデータを使用する事が自然である。

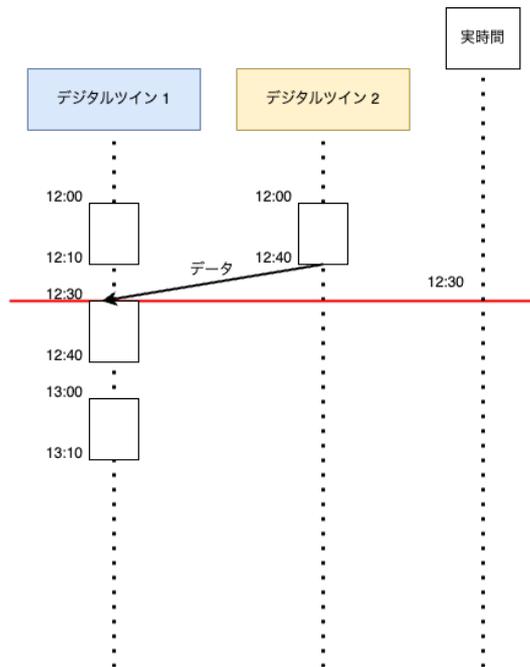


図 3.6: デジタルツイン間の依存関係 2

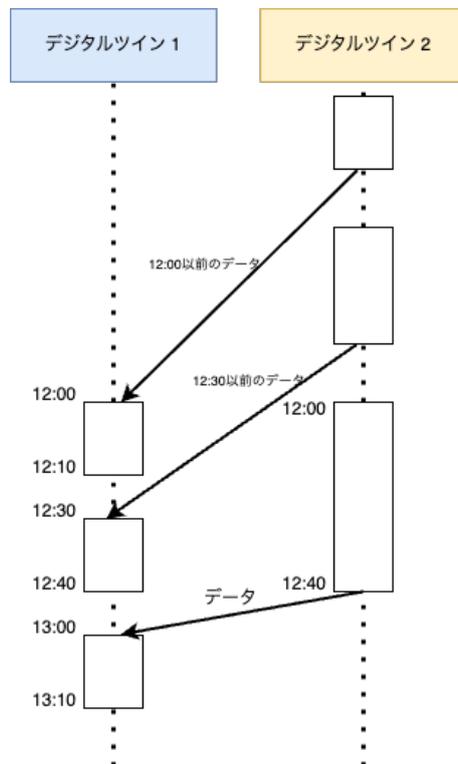


図 3.7: デジタルツイン間の依存関係 3

デジタルツイン統合の実現のためにはシミュレータ内の時間軸の違いとそれに

よって発生するデータの依存関係について考える必要がある。

### 3.3.3 再計算

デジタルツインは物理空間の情報をセンシングしている。入力されたセンシングデータによって、デジタルツイン内部のシミュレーションによって将来のデータを出力している。最新のセンシングデータによって、将来データに変更がある場合、デジタルツインは再計算を行う。図 3.8 はデジタルツイン単体が再計算を行った場合の時間軸の図を表している。図 3.8(a) は 12:00 の時点でセンサデータを取得して 15:00 までのシミュレーションを終えていたとする。図 3.8(b) は 14:00 の時点で新しいセンサデータを取得する。14:00 のセンサデータから 14:00 15:00 のシミュレーション結果が正しくない事がわかった。この場合デジタルツインは再計算を行い 14:00 15:00 の正しいデータを出力する。この一連の流れをデジタルツインの再計算と呼ぶ。

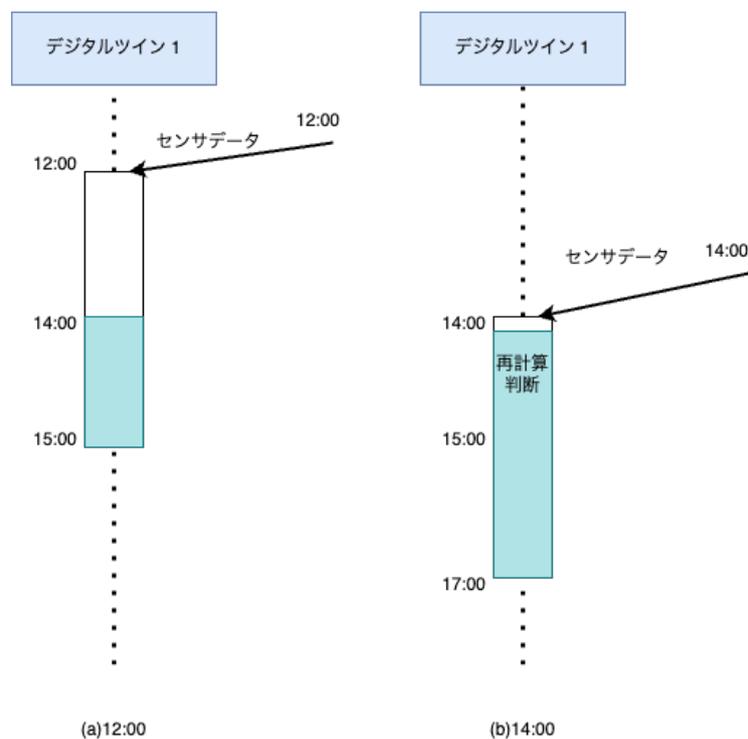


図 3.8: デジタルツインの再計算

再計算はデジタルツインの機能だが、デジタルツイン同士を連携するにあたって一度は他のデジタルツインに対して誤ったデータを送ってしまう可能性がある。

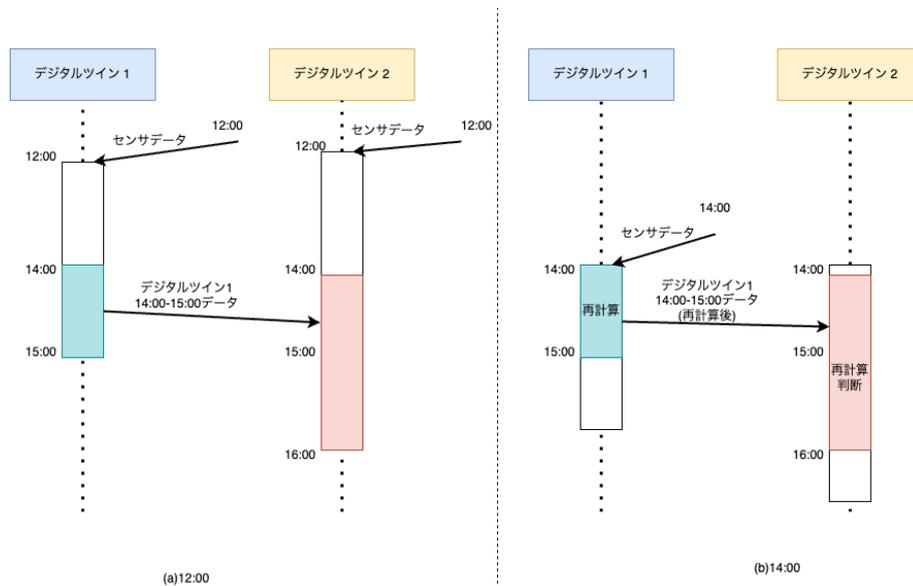


図 3.9: デジタルツイン連携時の再計算

図 3.9 はデータに依存関係があるデジタルツイン 1 とデジタルツイン 2 において、デジタルツイン 1 が再計算を行った場合の様子を表している。図 3.9(a) は、図 3.8 と同様にデジタルツイン 1 は 12:00 の時点で、15:00 までのシミュレーションを終えている。デジタルツイン 2 はデジタルツイン 1 の 15:00-16:00 までのデータを使用し 17:00 までのシミュレーションをおこなっている。図 3.9(b) では 14:00 時点で最新のセンサーデータがデジタルツイン 1 に入力され、再計算を行っている。この再計算によって 14:00-16:00 のデータが更新される。これを受けて、デジタルツイン 2 は再計算を行うかどうか判断しなければならない。

以上のことからデータに変更があった場合、対象のデジタルツインは、データの変更を知る機能と再計算とその判断機能を有する必要がある。

また、再計算を繰り返すことによって保存するデータが増えていく。

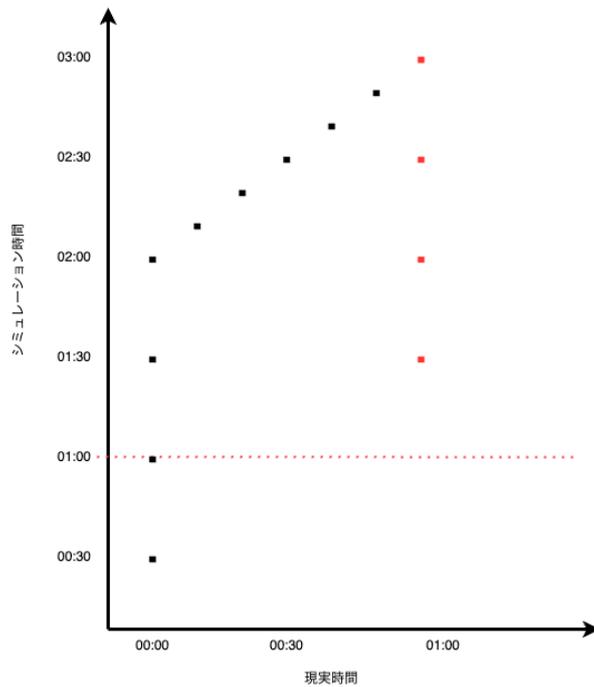


図 3.10: 再計算時のデータ生成

図 3.10 はデジタルツインのシミュレーション結果のデータを表している。0:00 に起動したデジタルツインは2時間先を30分間隔でシミュレーションすることができるとする。1:00 に再計算を行い新たにデータが生成されると、過去のデータと再計算したデータの両方を保持することになる。他のデジタルツインが過去のデータを使用している可能性があるため削除することができない。

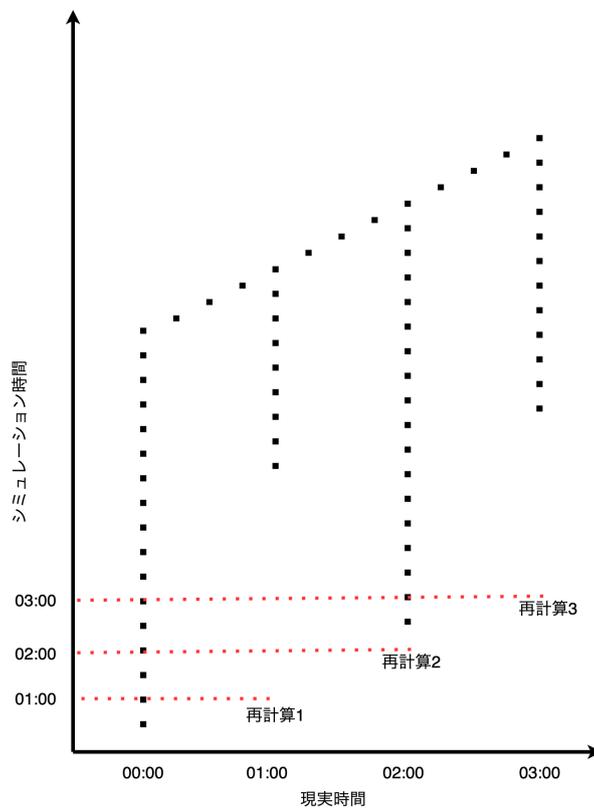


図 3.11: 再計算が複数回行われた時のデータ生成

図 3.11 はデジタルツインが再計算を 3 回行った時のデータの様子を表している。このように再計算を行う度に保持するデータが多くなってしまふ。そのため保持するデータを最小限にするような機能が求められる。

### 3.4 一般的な連携方法

複数のシミュレータを連携するとき、依存関係に矛盾を生まないためにバリア同期が用いられる。図 3.12 にバリア同期の様子を示す。

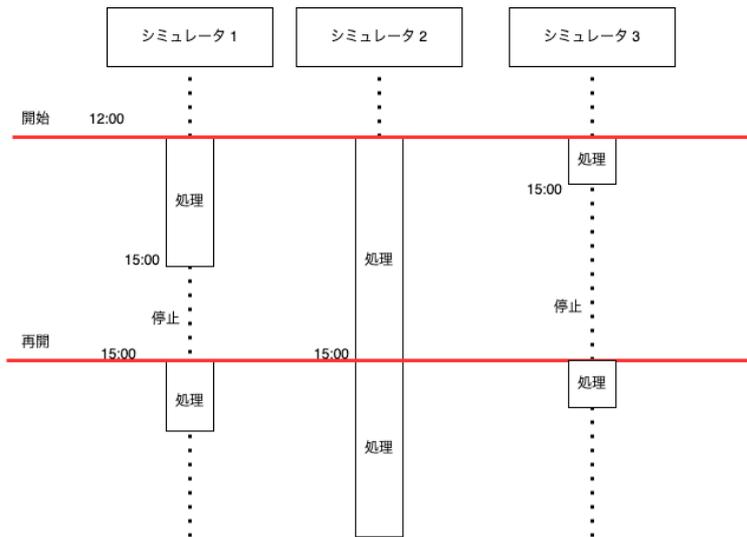


図 3.12: バリア同期

バリア同期は全体を矛盾なく進行させるために、処理の速いシミュレータは、処理が一番遅いシミュレータが完了するまで、停止などをして待ち同期を取る方法である。図 3.12 のシミュレータ 3 とシミュレータ 1 は処理を完了し、シミュレータ 2 の完了を待っている状態である。シミュレータ 2 の処理が完了したら、再びすべてのシミュレータが起動し、繰り返す。

バリア同期の利点としては、一番遅いプロセスに合わせて全体を停止させるため、データの矛盾が生まれにくい。しかし、極端に遅いプロセスが存在する場合、計算資源の利用効率が悪くなる。

本研究のデジタルツイン統合では 3.2 で述べたように、処理時間の違いや、3.3.1 での時間軸の違いから、計算資源を無駄にする可能性が高い。このようことから今回のデジタルツイン統合にバリア同期は適さないと考えた。デジタルツインはお互い別々の目的を持っており、お互いに疎な関係であるから必要な時のみ通信する仕組みの方が適切である。

# 第4章 デジタルツイン統合の提案 手法

前章3.3でデジタルツイン統合における課題をいくつか挙げた。時間軸の違いや、データの依存関係に矛盾という課題を解決する一つの手法としてバリア同期を挙げた。時間軸が大きく異なるデジタルツイン同士の連携をおこなった場合、バリア同期では資源を無駄にする可能性が高い。

本研究ではコーディネータによる連携管理と共有DBによるデータ管理を提案し、資源をより効率よく活用する。

## 4.1 提案する連携方法

3.4にて一般的にはバリア同期を行うことで連携することができる。しかし、本研究で想定しているデジタルツイン同士の時間の差は小さい場合もあれば大きい場合もある。一番遅いデジタルツインの時間に合わせて全体を停止させるのは効率的とは呼べない。そこで、デジタルツインの動き方と依存関係について以下の3つに分類する。

- 長時間の差が出る場合
- 短時間の差が出る場合
- 周期的に起動する場合

### 4.1.1 長時間の差が出る場合

デジタルツインを統合する上で、3.3.1より時間軸の違いからシミュレータ時間に大幅な差が生まれる事がわかった。そこで統合するデジタルツイン全体を管理する「コーディネータ」を設ける。コーディネータはデジタルツインに対して、起動、停止などのメッセージを送るアプリケーションである。

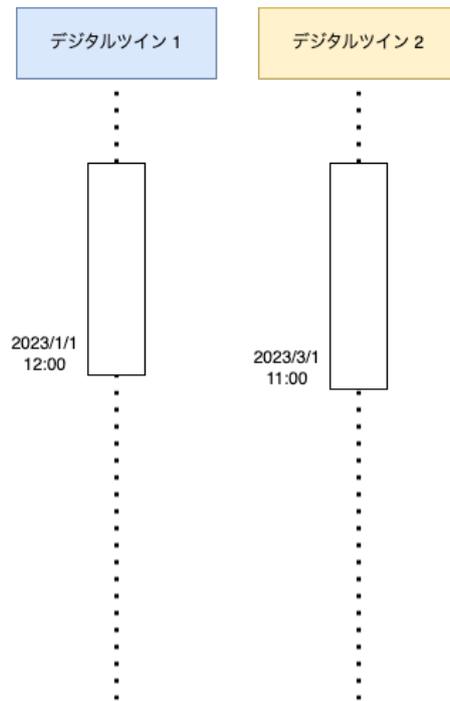


図 4.1: 極端な時間のずれ

図 4.1 は時間軸の違いから 2 つのシミュレータ時間が離れてしまっている例である。デジタルツイン 1 は 2023/1/1 のシミュレーションをおこなっているが、デジタルツイン 2 は 2023/3/1 のシミュレーションをおこなっている。時間が経つにつれてデジタルツイン 1 とデジタルツイン 2 の時間の差が広がる。差が大きくなると、お互いのデジタルツインの連携のメリットがなくなる。遠い将来のデータも過去のデータも、お互いのデジタルツインが使用しづらいからである。また、データの保存にも限りがある。

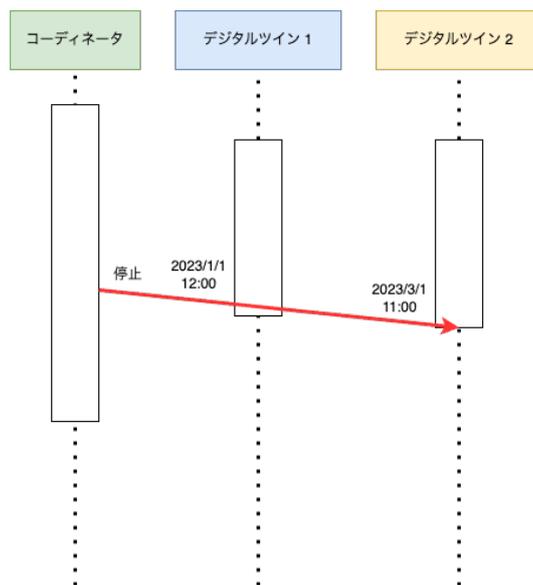


図 4.2: コーディネータによるデジタルツインの停止

そこで、デジタルツインの制御を担当するコーディネータを設ける 4.2. 他のデジタルツインと比べてシミュレーション時間が進みすぎていると判断した場合にコーディネータが対象のデジタルツインに対して、停止要求のメッセージを送る。シミュレーション時間が縮まると再び起動開始のメッセージを送り、実行する。

#### 4.1.2 短時間の差が出る場合

3.3.2 で述べたように、時間軸の違いによって交換したデータの因果関係を保持できない場合が生じる場合がある。これを共有の DB を利用することで解決する。データの因果関係はシミュレーション時間よりも未来のデータを使用してしまうことが原因である。共有データベースを用いることにより、デジタルツインは自身のシミュレーション時間より前に生成されたデータを使用することによって因果関係を保持したままシミュレーションを行う事ができる。

表 4.1: データに付加する内容

	データ名	タイムスタンプ	製作者	生成日時
例 1	天気	2023/01/02 13:00.00	デジタルツイン 1	2023/01/01 13:00.00
例 2	気温	2023/01/01 12:00.00	デジタルツイン 2	2023/01/01 12:00.00

実時間9:00

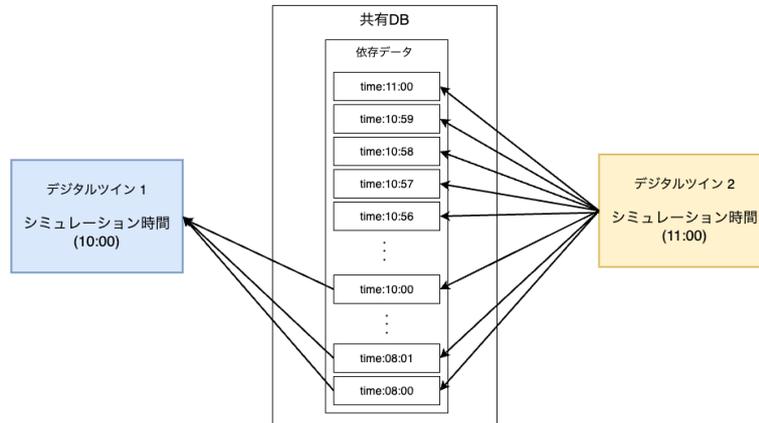


図 4.3: 短期的な時間の差によるデータの依存

表 4.1 は共有データベースに保存されているデータが持つべき情報と例を示している。データにシミュレーション時間のタイムスタンプを追加することで、他のデジタルツインが、未来のデータを参照することがないようにする (図 4.3)。

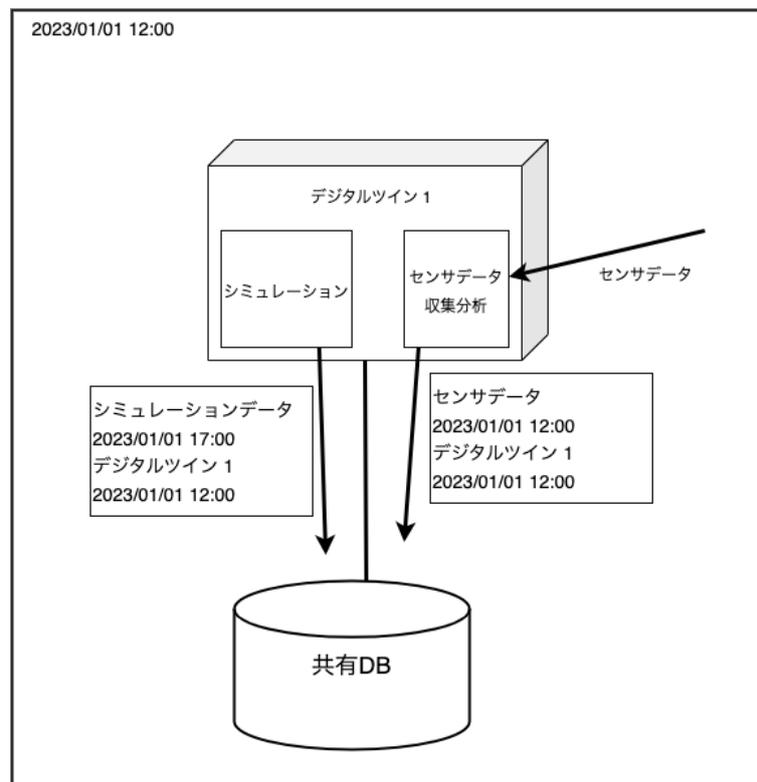


図 4.4: センサデータとシミュレーションデータの保存

図 4.4 はデジタルツインが共有データベースにデータを保存する時の様子を表し

ている。センサデータを保存する場合はタイムスタンプと作成日時が同じになる。製作者はセンサデータを収集しているデジタルツインとなる。また、シミュレーションデータを保存する場合、タイムスタンプはシミュレーション内での時間を記述する。

### 4.1.3 周期的に起動する場合

デジタルツインによってはシミュレーション自体を頻繁に行わず、周期的に起動してデータを生成する場合がある。この場合、常に起動させておくより、適当な時間にデジタルツインを起動させる方が資源を有効活用できる。そこで前節のコーディネータを用いて、起動タイミングを登録しておくことで、適当な時間にメッセージを受け取り、デジタルツインを実行する。起動するまでは停止しているため、その間の資源を削減することができる。

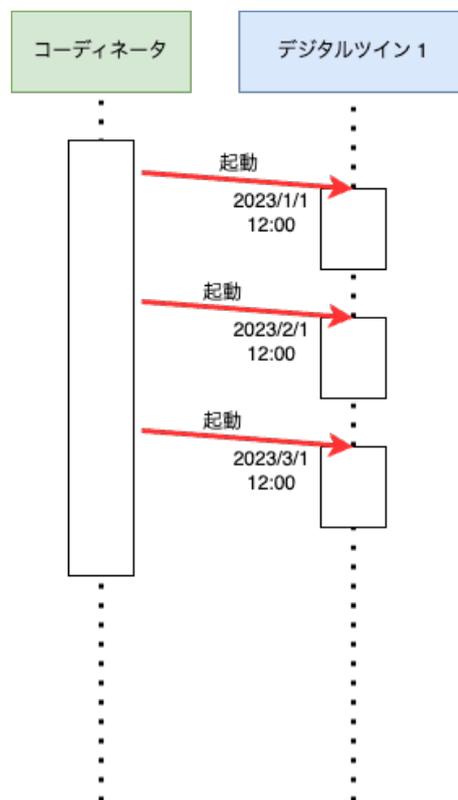


図 4.5: 周期的な起動

### 4.1.4 再計算のデータ管理

3.3.3で述べたように再計算を行うとデータの量が多くなっていく。そこで、2つのデジタルツインが使用しているデータについて考える。

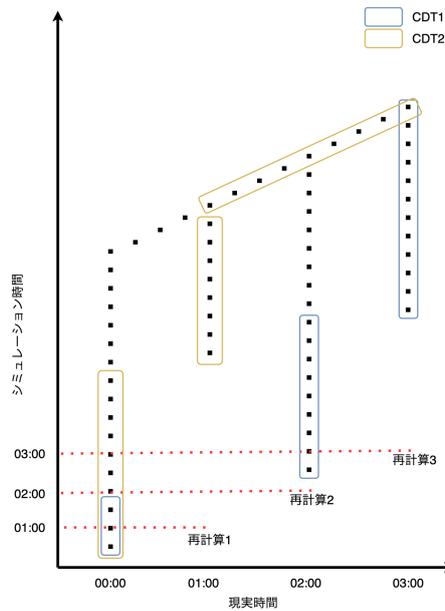


図 4.6: 再計算と使用するデータ

図 4.6 では CDT1 がデータを出力しており、CDT2 が使用している。CDT1 は再計算を行うと、再計算したデータをデジタルツインの出力結果とする。CDT2 は 2 回目の再計算の結果を使用し、1 回目と 3 回目の再計算のデータを使わないとすると、枠内のデータのみが使用するデータとなる。枠外のデータは CDT1 も CDT2 も使わないデータなため削除することで保存領域を広げることができる。そこで、使用しないデータにはラベル付を行い、すべてのデジタルツインのラベルが貼られていた時、データを削除する。なお、対象のデータを使用するデジタルツインの名前はコーディネータから取得する。

## 4.2 デジタルツイン開発者の基準

次にデジタルツイン開発者が考慮すべき基準について述べる。センサの更新が遅く、起動する回数が少ない場合は周期的な起動を行うのが適切である。次に計算頻度が高く、値を常に更新し続けるようなデジタルツインは DB に保存するデータ数を考慮するべきである。データを短期間しか保存しない場合はスリープする可能性がある。データに変化があまり起こらず、長期間にわたって保存できるのであれば、DB 内のデータのタイムスタンプでデータを利用することができる。どのデジタルツインが連携されるかはわからないため、適当な範囲のデータ利用とデータ保存が求められる。

# 第5章 システム提案

第3章にてデジタルツイン統合における課題をあげ、第4章にて、解決方法について述べた。本章では上述した課題を解決する具体的なシステムの提案を行う。

## 5.1 システム概要

図5.1はデジタルツインを統合するシステムの全体図である。

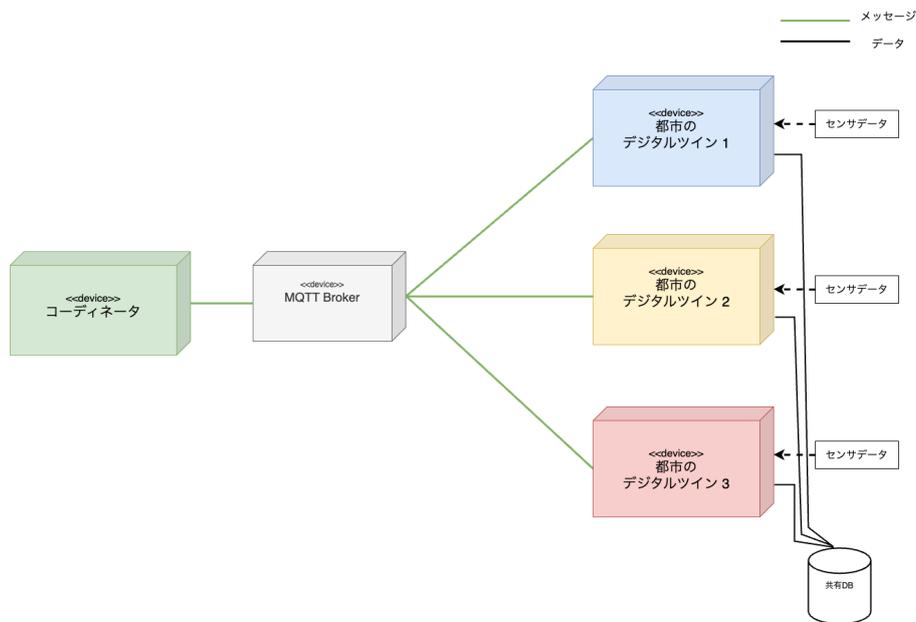


図 5.1: システム全体図

システムはデジタルツイン、コーディネータ、MQTT Broker、共有DB、センサで構成されている。提案するデジタル連携基盤の機能は大きく分けて以下の2つである。

1. メッセージによる制御
2. データベースによるデータ交換

図 5.1 のメッセージ線にてコーディネータ - デジタルツイン間, デジタルツイン - デジタルツイン間のメッセージ交換によって, 連携を実現する. 共有 DB にてデジタルツイン間でデータを共有する.

## 5.2 メッセージ交換

メッセージ交換には MQTT(Message Queuing Telemetry Transport)[23] を用いる. MQTT はクライアントとブローカーに分かれ, ブローカーを介してパブリッシュ・サブスクライブ方式でメッセージ交換をするプロトコルである. パブリッシャーはデータを生成したらトピックを指定して, ブローカに対してパブリッシュする. サブスクライバーは必要としているトピックを指定してサブスクライブすることによって, トピックが更新されたときに, サブスクライバーに配信されるようになる.

交換するメッセージは, コーディネータから配信される起動, 停止, デジタルツインから配信されるシミュレータ内時間, 再計算によるデータの更新である.

表 5.1: MQTT にて交換するメッセージ

トピック	メッセージの内容	方向
control/wakeup	・デジタルツインの ID	コーディネータ→デジタルツイン
control/sleep	・デジタルツインの ID	コーディネータ→デジタルツイン
cdt/time	・デジタルツインの ID ・シミュレーション時間 ・依存データ名と期間	デジタルツイン→コーディネータ
cdt/del	・デジタルツインの ID ・データ名 ・削除する期間	デジタルツイン→コーディネータ
cdt/update	・デジタルツインの ID ・生成時刻 ・データ名 ・更新した期間	デジタルツイン→デジタルツイン
cdt/info	・デジタルツインの ID ・依存するデータ名	デジタルツイン→コーディネータ
cdt/register	・デジタルツインの ID ・入力するデータ ・出力されるデータ ・状態 ・周期的な起動の有無	デジタルツイン→コーディネータ

表 5.1 はコーディネータ, デジタルツイン間でやり取りされるメッセージの一覧である.

次にデジタルツイン連携の流れについて説明する.

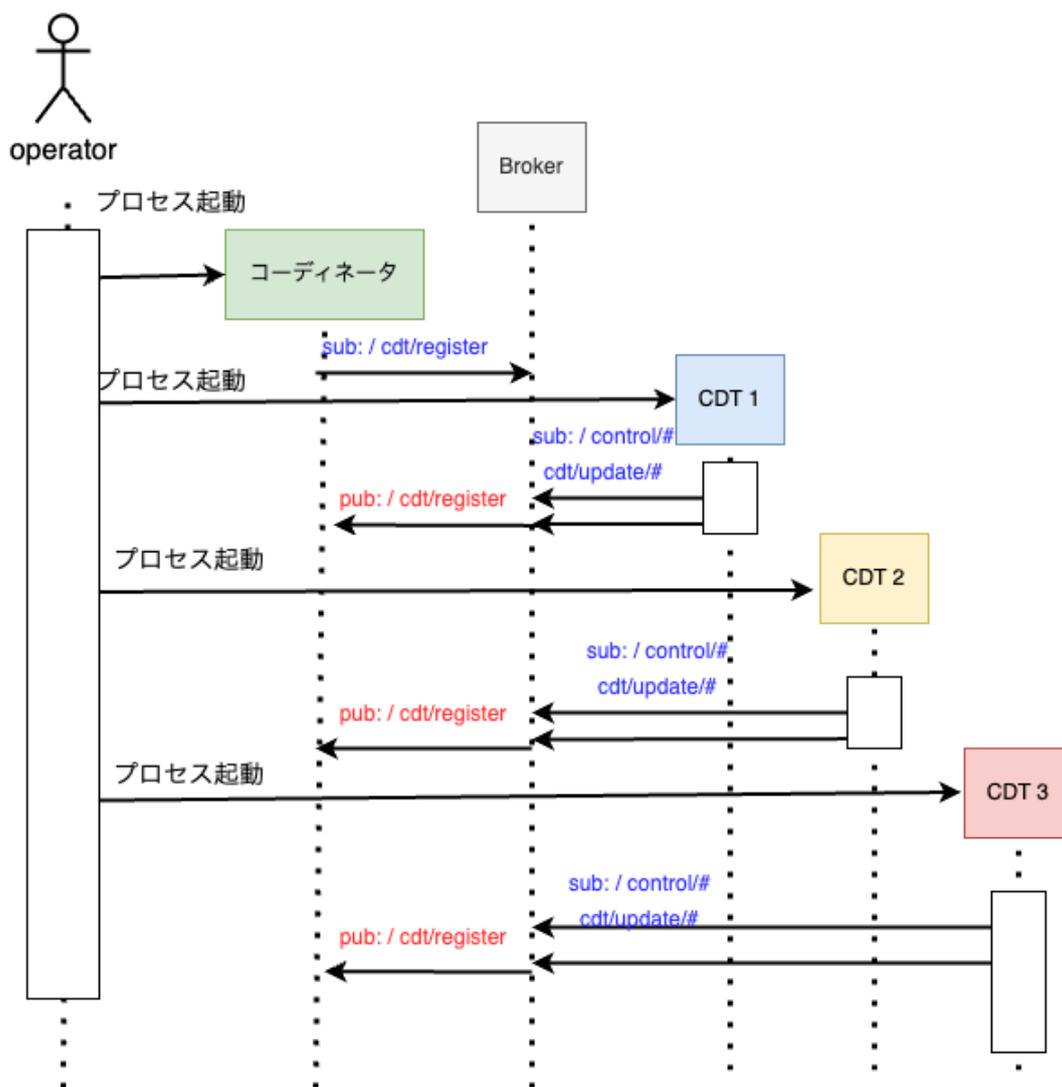


図 5.2: 連携初期メッセージ

図 5.2 はデジタルツイン連携の流れを表している. まずデジタルツイン連携を行いたいオペレータはコーディネータを起動させる. するとコーディネータは MQTT Broker に対して, 「cdt/#」に関するトピックをサブスクライブする. 次にオペレータは連携させたい都市デジタルツインを起動させる. 都市デジタルツインが起動するとまず, MQTT Broker に対して「control/#」と「cdt/update/#」に関するトピックをサブスクライブする. その後, 「cdt/register」のトピックで

デジタルツインの情報をメッセージに入れて publish を行う。コーディネータは「cdt/register」に関するトピックを購読しているので、連携に参加するデジタルツインを登録する。登録内容は、デジタルツインの ID, 自身が出力するデータ, 利用するデータ, 周期起動の有無である。

## 5.2.1 シミュレーション完了時のメッセージ

表 5.1 「cdt/time」のトピックはデジタルツインが1回のシミュレーションでデータを生成したときにデジタルツインがコーディネータに対して送るメッセージである。以下にメッセージの例を示す。

コード 5.1: シミュレーション完了時のメッセージ

```
1 {
2   "id":"community",
3   "datalist":[
4     {
5       "name":"weather",
6       "start":"2023-01-25 15:34:09.879619",
7       "end":"2023-01-25 16:01:09.879619"
8     }
9   ],
10  "cycle":"None"
11 }
12 }
```

メッセージには自身の ID と共有 DB 内で使用したデータとデータのタイムスタンプの期間を表している。

## 5.2.2 DB 内の過去データを削除するときのメッセージ

表 5.1 「cdt/del」のトピックは自身が生成しているデータを削除するときデジタルツインがコーディネータに対して送るメッセージである。

コード 5.2: データ削除時のメッセージ

```
1 {
2   "id":"community",
3   "name":"people",
4   "start":"2023-01-25 10:34:09.879619",
5   "end":"2023-01-25 12:01:09.879619"
6 }
```

自身の ID と削除したいデータ名と期間を記述する。

### 5.2.3 デジタルツインを停止させる時のメッセージ

表 5.1 「control/sleep」のトピックはコーディネータがデジタルツインに対してシミュレーションを停止させるときに用いるメッセージである。ペイロードはデジタルツインの ID 名である。5.2.2 でデジタルツインが削除したいデータと期間が、5.2.1 のデータ使用期間が重複する場合に、DB 内のデータを削除しようとしているデジタルツインに対して停止のメッセージを送る。

### 5.2.4 停止しているデジタルツインを起動させるときのメッセージ

5.2.3 にてデジタルツインが停止する際の動作の説明を行った。5.2.1 で述べたようにデジタルツインはシミュレーションが完了すると、使用したデータとそのタイムスタンプの期間をコーディネータに伝える。この期間と停止しているデジタルツインが削除しようとしている期間が重複しなくなったときに、コーディネータはデジタルツインに足して起動のメッセージを送る。メッセージは表 5.1 の「cdt/wakeup」のトピックでペイロードはデジタルツインの ID 名である。

### 5.2.5 デジタルツインが再計算を行った時のメッセージ

3.3.3 で述べたようにデジタルツインはセンサデータが更新されると、再計算を行う可能性がある。デジタルツインの出力結果を入力としているデジタルツインが存在する場合、値が更新されたことを通知する必要がある。そこで、再計算を行ったデジタルツインは「cdt/update」のトピックを用いて再計算を行ったことを他のデジタルツインに対して通知する。その時のメッセージ例を以下に示す。

コード 5.3: 再計算時のメッセージ

```
1 {  
2   "id":"community",  
3   "name":"people",  
4   "date":"2023-01-25 13:01:09.879619",  
5   "start":"2023-01-25 14:10:09.822329",  
6   "end":"2023-01-25 15:01:10.879619"  
7 }
```

上記のメッセージは community デジタルツインがデータ名 people の”start” ”end”のタイムスタンプの期間のデータが更新されたことを表している。対象のデータを使用しているデジタルツインはこのメッセージを受けて、再計算を行うか判断する必要がある。

## 5.3 データ保存

4.1.2にて共有データベースを使ったデータ交換を提案した。以下に json 形式で保存したデータの例を示す。

コード 5.4: 保存データ例

```
1 {
2   _id: ObjectId('63d0cd61687dd82063cf7047'),
3   id: 'community',
4   date: '2023-01-25 15:34:09.879619',
5   time: '2023-01-25 15:34:59.649315',
6   nouse: [],
7   log: 'com_sen2023-01-25 15:34:09.649315community2023-01-25
      15:34:09.879619',
8   name: 'people',
9   data: {
10
11     ...
12
13   }
14 }
```

データの保存は json 形式で行う。id はデータを生成したデジタルツインの ID を入れる。date では生成した実際の時間, time はシミュレーションの中での時間を表してる。name はデータ名である。nouse はアップデートなどで使用しなくなったデータにラベリングするために設けてある。

次にデータベースの構造について、MongoDB を使った例を図 5.3 に示す。

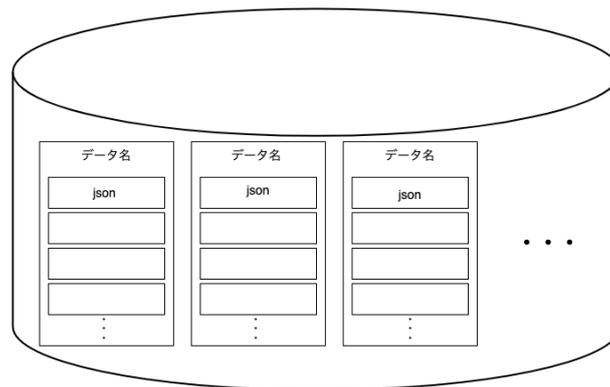


図 5.3: MongoDB を使ったデータベース構造

mongoDB はドキュメント指向型の NoSQL データベースである。複数のデジタルツインを統合する上で、データ構造を一致させる必要がある。データ構造に柔軟に対応するためにドキュメント指向である MongoDB を選択した。データごとにコ

レクシオンを作成し、コード 5.1 で示したようなドキュメントとしてデータを保存する。4.1.1 で提案したように、デジタルツインはデータを参照する際、ドキュメントに記載されているタイムスタンプ(シミュレーション時間)に基づいて検索を行うため、未来のデータを入力することはなくなる。

## 5.4 起動管理

### 5.4.1 停止と起動

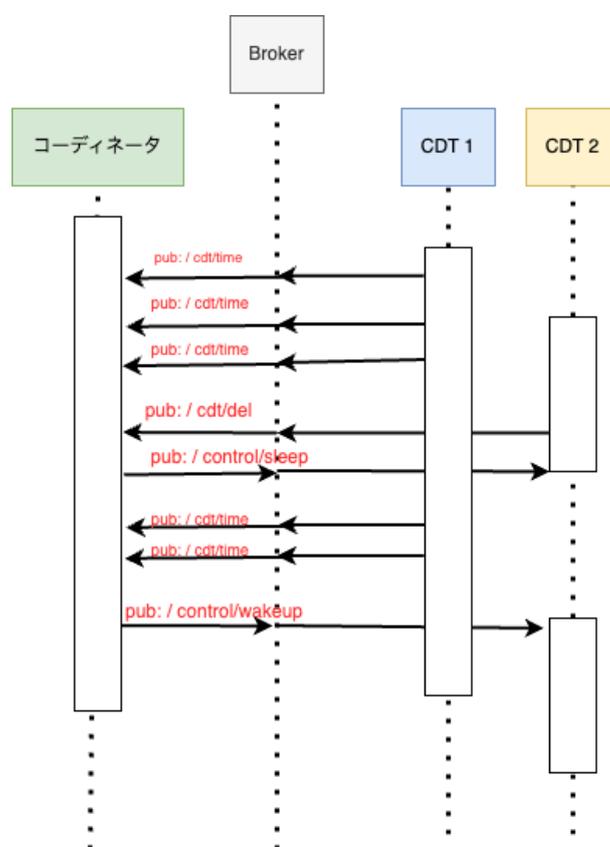


図 5.4: 長期的な時間の差

5.2 コーディネータによってデジタルツインの起動管理を行えることを説明した。長期的な時間の差が生じた場合について説明する。大まかな流れは5.2.1 - 5.2.4で説明している。デジタルツインはシミュレーションを行い、データを保存するタイミングで、コーディネータに対してどのデータのどの時間を使用したかを MQTT の「cdt/time」のトピックで publish する(図 5.4 の CDT1)。次に CDT2 がデータベースの古いドキュメントを削除する場合を考える。コーディネータに対して「cdt/del」のトピックで、データ名と該当する時間を報告する。コーディネータは

他のデジタルツインがこの期間のデータを使用していないか判断し、使用しているデジタルツインが存在すれば「control/sleep」のトピックにて CDT2 を停止させる。該当のデータが他のデジタルツインで使用されなくなると判断されれば「control/wakeup」にてデジタルツインを起動させる。

## 5.4.2 周期的な起動

次に周期的な起動をするデジタルツインについて説明する。図 5.2 にてデジタルツインは起動と同時にコーディネータに対して自身の登録を行う。登録情報に周期に関する記述を行う (コード 5.2)。

コード 5.5: 周期的な起動を行うデジタルツインの設定

```
1 SIM_INFO = {  
2     ...  
3  
4     "cycle":{  
5         "y/n":"y",  
6         "term":"10",  
7         "start":"None"  
8     }  
9 }
```

「y/n」でデジタルツインが周期的な起動を行うか選択することができる。「term」にて何秒間隔で起動メッセージを配信してもらうか指定する。「start」で記述した時間から周期的に配信してもらうことができる。「None」の場合はコーディネータに登録した時間から周期的な配信が始まる。

## 第6章 実験

第5にてコーディネータとMQTTによるメッセージ交換と共有DBについて提案した。本章では3つの都市デジタルツインを提案しデジタルツイン連携の実験を行う。

### 6.1 実験対象

3.2にてシミュレータを性質ごとに分類した。異なる性質のデジタルツインとして、「公民館デジタルツイン」、「防災デジタルツイン」、「都市計画デジタルツイン」の3つを例に挙げる。

日本が抱える問題として、地方の過疎化が挙げられる。市町村では公民館などの公共施設の維持が困難になり統廃合などを行うケースがある。そこで、公民館の運営指針を決める1つの手段としてデジタルツインを用いる。デジタルツインを用いて、公民館内にいる人の行動のシミュレーションを行う。

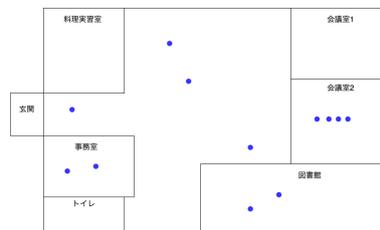


図 6.1: 公民館デジタルツインの例

図6.1は公民館のフロアマップと青点で人の位置を表したものとなっている。MASを使って、青点の人間をエージェントとしてルールを持たせることで公民館内をどのように人が動き回るかシミュレーションすることができる。これに加えて測位センサを用いることで、実際の人間の位置を確認することができる。

公民館デジタルツインでは実際の人々の位置とMASによる人の動きの予測をするデジタルツインとなっている。

表 6.1: 公民館デジタルツインの入出力

入力	出力
センサデータ (測位)	施設の人数データ (来訪者数)
イベントデータ	人の位置データ
天候データ	施設利用集計データ

表 6.1 に公民館デジタルツインの入出力を示す。測位データによって公民館に来館している人の位置を割り出す。そこからイベントデータと天候データを入力として、1 時間後までの人の動きを予測する。出力されるデータは来訪者数である人数データ、人の位置データである。施設利用集計データは 1ヶ月おきに出力される集計データである。

次に防災デジタルツインについて述べる。防災デジタルツインは街に複数個設置した積雪センサと天候センサにより、積雪状態を確認し、除雪車の除雪ルートを出力するシミュレーションとなっている。なお、センサは公民館に設置されているものとする。

表 6.2: 防災デジタルツインの入出力

入力	出力
センサデータ (積雪, 天候)	天候データ
建物データ	除雪ルート
地理データ	
施設の人数データ	

表 6.2 に防災デジタルツインの入出力を示す。センサデータによって今後の天気や積雪状況を予測し、除雪車のルートを出力する。市民に多く使われている道路や施設を優先して出力するため、施設の人数データも入力として利用する。

都市のデジタルツインは、政府などが公表する国勢調査や、住民基本台帳、施設の利用データなどをもとに 10 年単位で 40 年先の人口動態を計算するものである。

表 6.3: 都市計画デジタルツインの入出力

入力	出力
人口動態調査	人口動態
国土地理数値情報 (施設データ)	施設存続状況
施設利用集計データ	

表 6.2 は都市計画デジタルツインの入出力を表している。人口動態調査をもとに今後の人口推移を予測し、人口動態を出力する。また、人口に基づいて、施設の存続状況を予測する。

## 6.2 実験概要

### 6.2.1 構築したシステム全体図

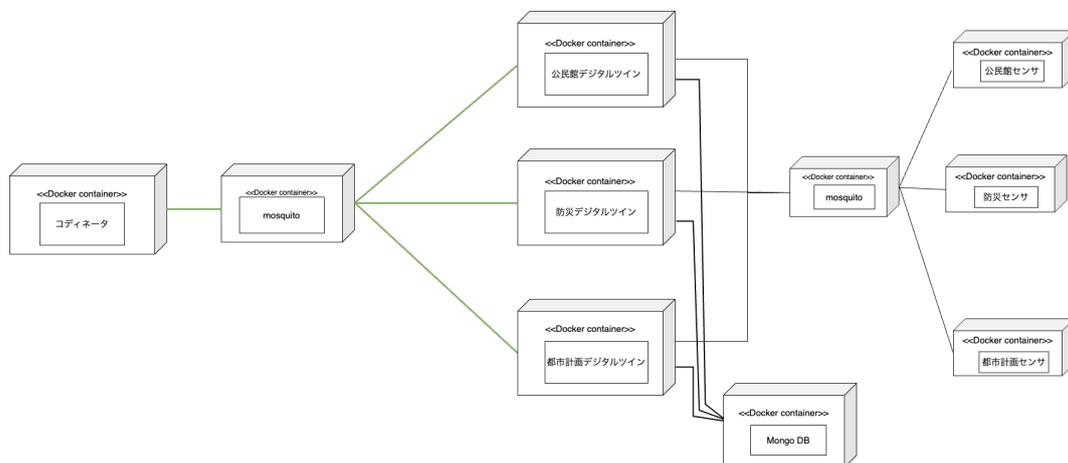


図 6.2: 実験でを使用したシステム全体図

図 6.2 に本実験でを使用したシステム全体図を示す。Docker を用いて、コーディネータ、mosquitto MQTT BROKER、デジタルツイン、センサ、MongoDB を構築した。センサ - デジタルツイン間はセンサデータ、コーディネータ - デジタルツイン間は制御などのメッセージ、デジタルツイン - MongoDB 間はデータが流れる。

### 6.2.2 実験環境

次に実験を行った環境について述べる。以下の表 6.4, 6.5, 6.6 の通りとなっている。

表 6.4: 実験環境

OS	Ubuntu 20.04.1 LTS
CPU	Intel Core i5 8259U 2.3Ghz
メモリ	DDR4 16GB

表 6.5: docker イメージ

Mongo DB	version 4.2.5
Ubuntu	version 20.04
mosquito	version 2.0

表 6.6: python とライブラリ

python	version 3.8
paho-mqtt	version 1.6.1
pymongo	version 3.3.1

### 6.2.3 デジタルツインの設定と連携方法

本実験ではデジタルツインのシミュレーションは行わず、入出力のみに着目する。表 6.7 のデジタルツインのセンサと表 6.9 の入力データから表 6.8 のデータを出力する。

表 6.7: センサと取得間隔

デジタルツイン	センサデータ	サイズ	取得間隔
公民館	測位データ	3KB	500ms
防災	気候データ	900KB	10m
都市計画	人口動態総覧	2.8MB	30m

表 6.7 はセンサデータとデータ 1 つあたりのサイズ、取得間隔を表している。例えば公民館デジタルツインは 500ms の間隔で 3KB の測位データを受信する。

表 6.8: 出力データ

デジタルツイン	データ	サイズ	データ数	シミュレーション期間	出力間隔
公民館	施設の人数データ	1.4KB	720	1 時間先まで	10s
防災	気候データ	14KB	12	2 時間先まで	7s
都市計画	人口動態	2.8MB	5	40 年先まで	30m

表 6.8 はデジタルツインが出力するデータを表している。例えば防災デジタルツインは現在時刻から 2 時間先までのシミュレーションすることができる。12 データ生成されるので、30 分おきのデータが出力される。

表 6.9: 依存する入力データ

デジタルツイン	データ	生成元	使用範囲
公民館	気候データ	防災	計算している時間の 1 時間前
防災	施設の人数データ	公民館	計算している時間の 10 分前
都市計画	施設利用集計データ	公民館	最新のデータ

表 6.9 はデジタルツインが他のデジタルツインが出力するデータを使用する場合の使用範囲を表している。例えば、公民館デジタルツインは防災デジタルツインが出力する

力している気候データを使用している。公民館デジタルツインはシミュレータ時間の1時間前までのデータを使用する。例えば、現在時刻が10:00として、10:20分のシミュレーションを行う場合は9:20分までの気候データを利用することになる。

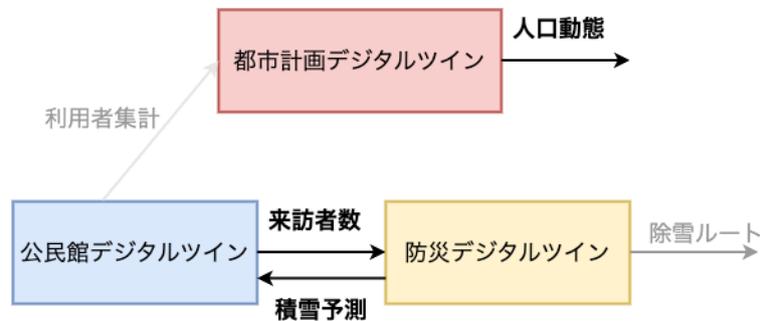


図 6.3: 想定するデジタルツインの依存関係

公民館デジタルツインと防災デジタルツインは短期的な時間差であればDBのタイムスタンプを参照する。長期的な差が生まれる場合にはコーディネータが停止、起動を行う。都市計画デジタルツインは周期的な起動をする設定をコーディネータに登録する。

## 6.3 実験内容

本実験では設定したデジタルツインを以下のパラメータを変更して実行した時のシミュレーションのドキュメント数を記録する。なお、本実験でシミュレータは計算を行わなわないため、再計算を行う判断基準を持っていない。再計算を行う確率を3%とし実験する。

- 過去ドキュメントの上限  
DBの保存領域には限りがある。そのため過去のドキュメントは一定期間が経つと削除する必要がある。過去ドキュメントの上限を設定することで、デジタルツインが過去の依存データにアクセスできなくなるため、コーディネータによってデジタルツインが起動管理される。

最初に公民館デジタルツインと防災デジタルツインについて、連携させずに起動し、ドキュメント数を記録する。次に上記のパラメータを設定して、「短期的な時間の差の連携」と「長期的な時間の差の連携」を実現し評価する。

## 6.4 実験結果

### 6.4.1 実験 1: 単体起動

デジタルツインの連携は行わず、単体でデジタルツインを起動させた時の様子を以下に示す。

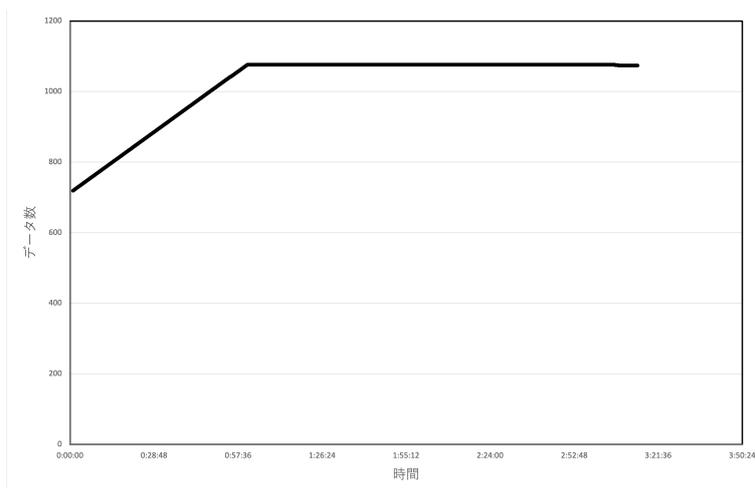


図 6.4: 公民館デジタルツインの単体起動

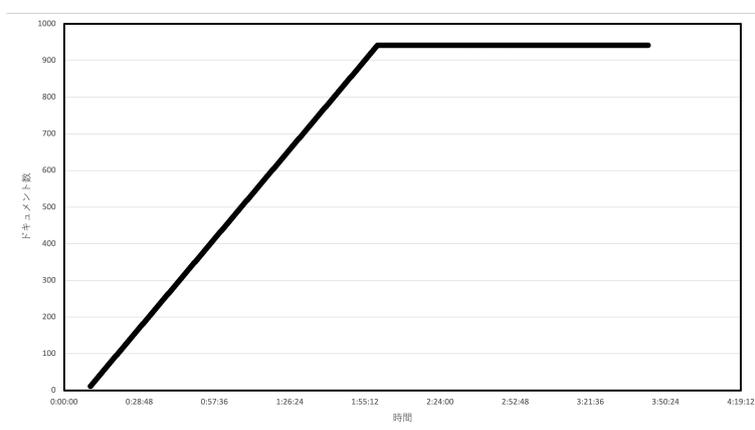


図 6.5: 防災デジタルツインの単体起動

図 6.4 と図 6.5 はそれぞれ、公民館デジタルツインと防災デジタルツインを単体起動させた時の、シミュレーションしたデータの数(タイムスタンプが現在時刻よりも未来のドキュメント)を表している。公民館デジタルツインのドキュメント数の最大値は1017、防災デジタルツインのドキュメントの最大値は947であった。つまり、公民館デジタルツインは1時間分のシミュレーションにて最大で約1000個、

最小で12個のドキュメントが生成される。防災デジタルツインは2時間分のシミュレーションにて最大で約1000個、最小で720個のドキュメントが生成される。

### 6.4.2 実験 2: 短期的な時間差と周期起動するデジタルツインの連携起動

表 6.9 で設定したように、公民館デジタルツインは1時間前の防災でデジタルツインのデータを使用する。つまり、防災デジタルツインの過去の500ドキュメントを使用する。公民館デジタルツインと防災デジタルツインが短期的な時間差になるために、防災デジタルツインのドキュメントの上限を1000に設定した。

表 6.10: 短期的な時間差と周期起動するデジタルツインの設定

デジタルツイン	ドキュメント上限	削除する数
公民館	1000	100
防災	1000	100
都市計画	60	12

3つのデジタルツインのドキュメント数について以下に示す。

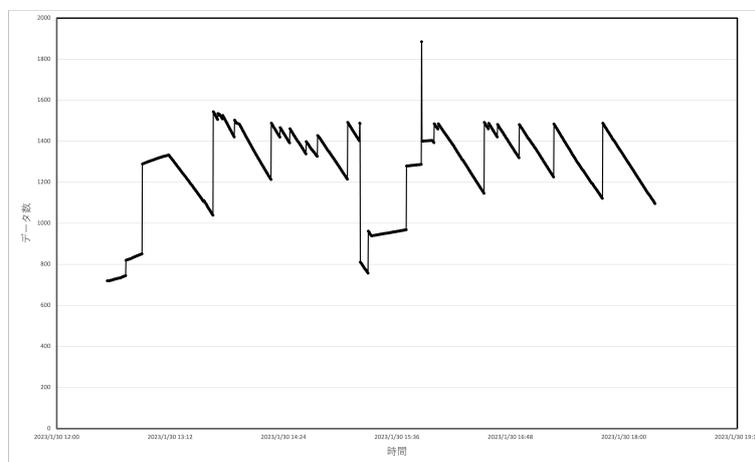


図 6.6: 短期的な時間差で動く公民館デジタルツインのドキュメント数

図 6.6 は公民館デジタルツインにおいて、タイムスタンプが現在時刻よりも先のドキュメントの数を表している。急にドキュメント数が増えるのは公民館デジタルツインが再計算を行った際に新たに生成したドキュメントが増えたからである。

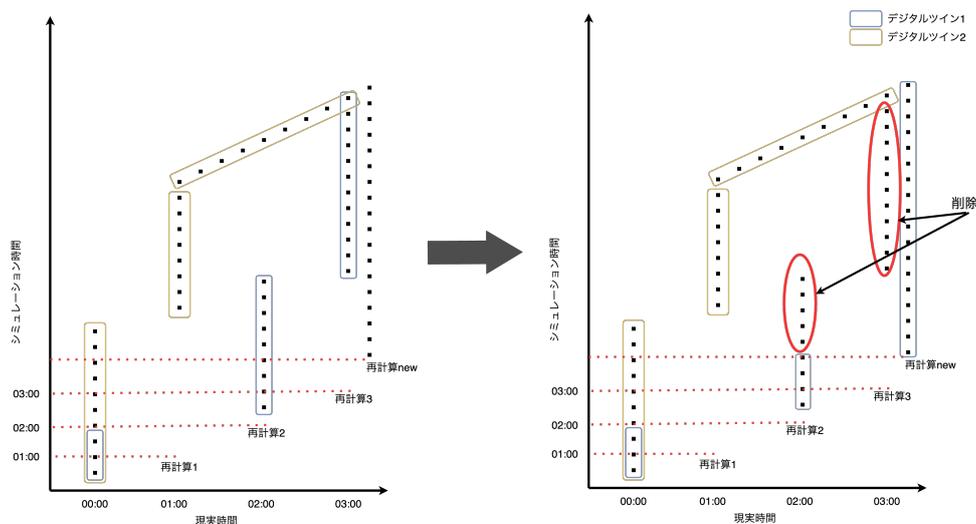


図 6.7: 再計算による削除

図 6.7 はデジタルツイン 1 が再計算を行った際のデータ数の様子を表している。デジタルツイン 2 は再計算の結果を使用しない値場合は図 6.7 右のようなデータが削除される。この削除されるデータをカウントしているため急激にデータ数が上下している。

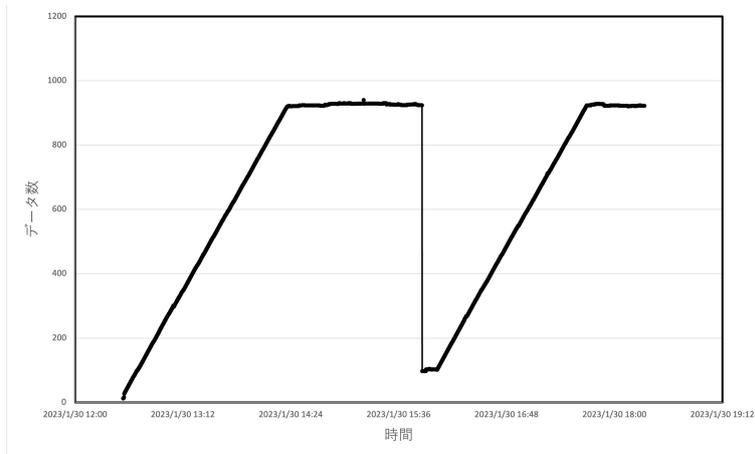


図 6.8: 短期的な時間差で動く防災デジタルツインのドキュメント数

図 6.8 は防災デジタルツインにおいて、タイムスタンプが現在時刻よりも先のドキュメントの数を表している。ドキュメントが急激に減少しているのは防災デジタルツインが再計算を行った際、公民館デジタルツインも再計算の結果を受け入れ、今までのドキュメントを破棄したからである。

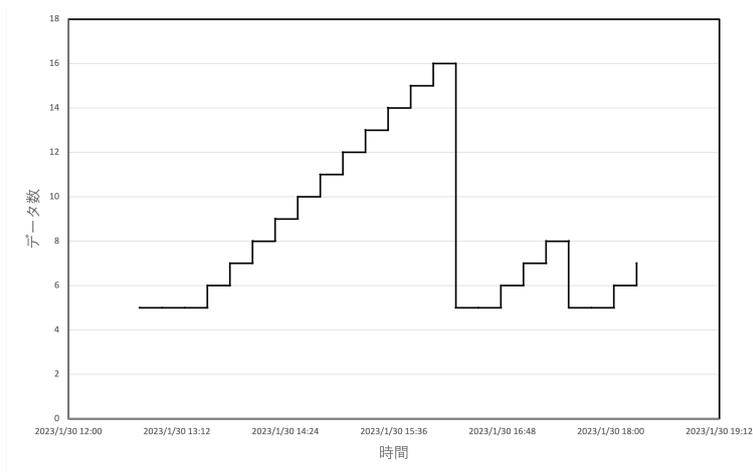


図 6.9: 都市計画デジタルツインの周期起動

図 6.9 は都市計画デジタルツインにおいてタイムスタンプが現在時刻より先のドキュメントの数を表している。

### 6.4.3 実験 3: 長期的な時間差と周期起動するデジタルツインの連携起動

次に防災デジタルツインが保持する過去のドキュメント数を 500 にし, 公民館デジタルツインが使用しているデータと, 防災デジタルツインが削除しようとしているデータが重複するように表 6.11 に設定した。

表 6.11: 長期的な時間差と周期起動するデジタルツインの設定

デジタルツイン	ドキュメント上限	削除する数
公民館	500	50
防災	500	50
都市計画	60	12

3つのデジタルツインのドキュメント数について以下に示す。

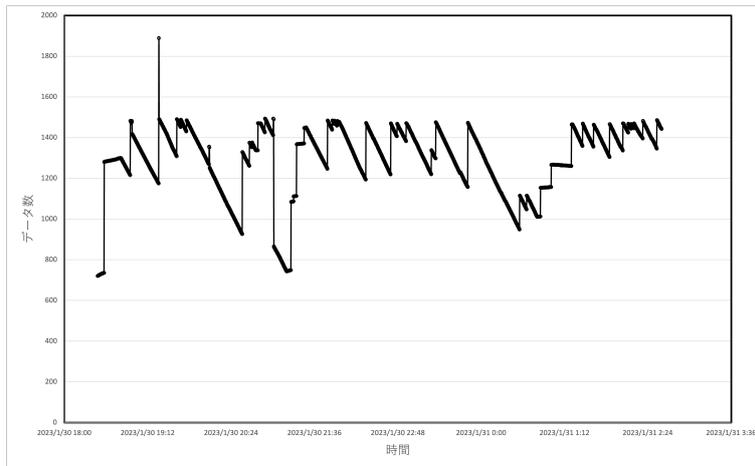


図 6.10: 長期的な時間差で動く公民館デジタルツインのドキュメント数

図 6.10 は公民館デジタルツインのドキュメント数を表している。削除するドキュメントと他のデジタルツインが使用するドキュメントに重複がないため図 6.6 と概ね同じ形をしている。

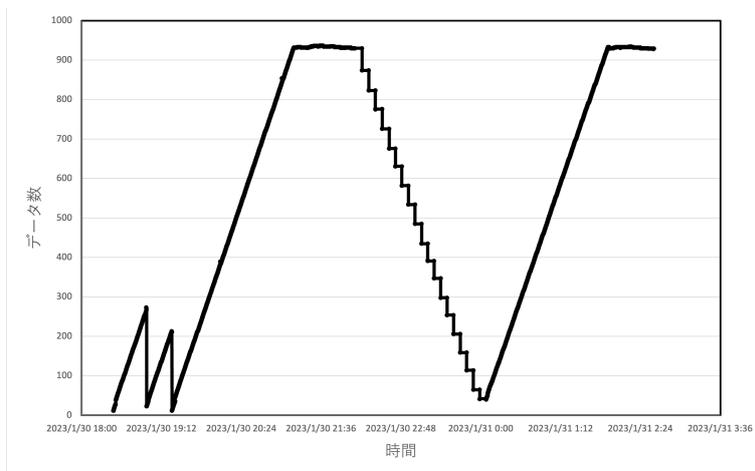


図 6.11: 長期的な時間差で動く防災デジタルツインのドキュメント数

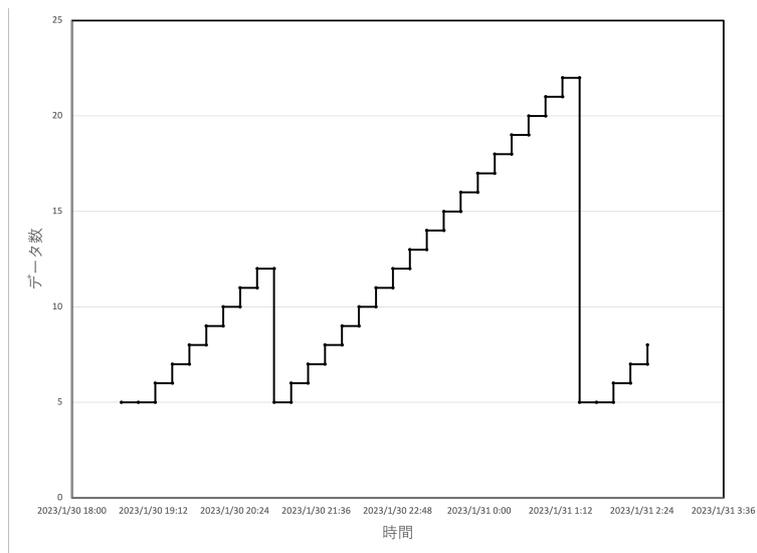


図 6.12: 都市計画デジタルツインの周期起動 2

図 6.11 に注目するとシミュレーションが途中止まっていることがわかる。これは防災デジタルツインが削除しようとしたデータを公民館デジタルツインが使用しているからである。なお、停止している時間は1時間 53分 45秒であった。

# 第7章 評価

## 7.1 実験に対する評価

### 7.1.1 データリソースについての評価

3.3.3でも述べたように再計算を行うと、過去のデータと再計算によって得られたデータの2種類が存在するようになる。他のデジタルツインが過去のデータを採用するか、再計算の結果を採用するかわからないため、再計算前のデータも保存しておくことになる。再計算の回数が増えると膨大なデータを保存しなければならなくなる。今回提案した連携手法によってデータ使用者の把握し、データを最適化することに成功した。表7.1と表7.2に再計算を行った回数と、ラベルによって削除したドキュメント数と、生成した総ドキュメント数を示す。

表 7.1: 実験2での再計算

デジタルツイン	再計算回数	ラベルによる削除	生成ドキュメント数
公民館	26	6115	14876
防災	107	3875	6003

表 7.2: 実験3での再計算

デジタルツイン	再計算回数	ラベルによる削除	生成ドキュメント数
公民館	46	14449	27001
防災	99	5232	7989

表7.3に再計算にすべてのデータを保存した場合と本実験でどのくらいリソースを削減できたか示す。

表 7.3: 削減したドキュメントの割合

デジタルツイン	実験2	実験3
公民館	41.10%	53.51%
防災	64.55%	65.49%

表 7.3 にある通り、公民館デジタルツインは約 45%、防災デジタルツインはやく 65%のドキュメントを削減することができた。

### 7.1.2 連携についての評価

表 7.4: 本研究の連携とバリア同期

	本研究の連携	バリア同期	本研究の連携/バリア同期
sleep 時間 (分)	113	168	67.26%
起動時間 (分)	223	168	132.73%

表 7.4 は実験 2 の結果とバリア同期を使用した場合の比較である。バリア同期は連携するシミュレーションの時間を同期するため、一番遅いシミュレーションに合わせる。今回の場合は公民デジタルツインの時間の進みが一番遅いため、防災デジタルツインの起動時間は 1/2 になる。実験 2 ではバリア同期よりも約 32%の計算資源を活用できていることがわかる。しかし、過去データの保存期間が極端に短いと、ほとんど起動しなくなる可能性がある。

次に「周期的な起動をする連携」について都市計画デジタルツインを扱った。コーディネータに起動周期を登録することで、周期的に起動通知を送信する。このデジタルツインについてもバリア同期を用いると一度起動してから 40 年経つまで起動しなくなる。そのため本研究での連携の方が優れているといえる。

## 7.2 課題に対する評価

本研究では、デジタルツイン統合における課題として「デジタルツイン内の時間の違いと依存データ使用における時間的矛盾」と「再計算によるデータの更新」であった。この課題に対し、「コーディネータとメッセージによるデジタルツイン制御」と「共有 DB とタイムスタンプ」を提案し実験評価を行った。十分な保存領域がある場合 (短期的な時間差がある場合) は実験 2 において、共有 DB を用いることでデータの時間的矛盾を生じさせないようにした。十分な保存領域がない場合 (長期的な時間差がある場合) は実験 3 において、コーディネータが防災デジタルツインを停止させることで、使用中のデータを削除することなく連携することが可能になった。周期的な起動をする場合は実験 2 の都市計画デジタルツインにおいて、コーディネータに起動周期を登録することによって周期的な起動管理が行えるようになった。最後に再計算によるデータの更新は実験 2,3 において、再計算メッセージを配信することで他のデジタルツインが再計算を把握することが可能になった。

## 7.3 議論

### デジタルツイン開発者にかかる負担

次に本研究の連携基盤に参加するデジタルツイン開発者にかかる負担について考える。

- データモデル, 構造の統一  
本連携基盤ではデジタルツイン同士がお互いにデータの意味を理解できなくてはならない。そのため統一されたデータモデルと構造を用いる必要がある。また, センサデータもデータ連携に用いるためにデータを変換する機能をつけなくてはならない。
- 依存データによる再計算判断機能  
他のデジタルツインで生成されたデータを使用する場合は, その値が更新される可能性がある。更新されたデータを適用するか, 過去のデータを利用するか判断する機能が必要になる。この機能は使用するデータの数だけ必要になる可能性がある。
- 各種メッセージとコーディネータ連携  
コーディネータに登録するための登録情報やメッセージ内容にはルールがある。ルールに従わないと正しく連携できない可能性がある。

## 第8章 終わりに

### 結論

本研究では異なる性質のデジタルツインを統合するための連携手法について提案した。

一般的な連携方法として、時刻を同期しながら進めるバリア同期が存在するが、時間同期を行うため、時間の進み方が大幅に違うデジタルツインを連携すると計算リソースに大幅な無駄が生じる。そこで、時間の進み方や性質の異なるデジタルツインが連携できるシステムが求められている。連携実現のためにデジタルツインを管理するコーディネータとデジタルツインのデータを管理する共有DBを提案した。コーディネータのメッセージによるデジタルツインの起動停止によって、起動方法の異なるデジタルツインを扱えるようになった。また、データにタイムスタンプを用いることで、データの因果関係に矛盾が生じるのをなくした。再計算によって新たに生成されたデータと過去のデータについて使用するかどうかラベル付を行うことで、データベースに保存するデータの量を最小限に抑えることができる。

デジタルツイン設計者は開発のルールに従うことで連携を行うことができる。実験では3つのデジタルツインを想定し、連携をおこなった。コーディネータによる周期的な起動や、デジタルツインの停止を実現することができた。また再計算を行った際にデジタルツイン同士がメッセージを交換することによって適切なデータ選択を行うことができ、全てのデータを保存した時と比べて40 - 65%のデータを削減することができた。

### 課題

今回の研究ではデジタルツインを連携するための最小限の設計となっている。また、シミュレーション結果の妥当性については評価していない。次は実際のシミュレータを用いて適切にデータが扱われ、シミュレーションできているか確かめる必要がある。また、データベースに関して簡単なシステムを提案したが、データの種類や量が膨大になった時のためにデータのアクセス速度を考慮した設計を行う必要がある。本研究では依存データを最小限にした。これは異なる目的のデジタルツイン同士の連携のため、密接に関わるようなデータは少ないという理由である。しかし、複数の大量のデータを取り扱った時の連携についても考慮するべきである。また、データモデルの統一は課題の一つである。都市OSのプラットフォームであるFIWAREではNGSI-LDを利用して情報モデルを記述している。今回提案したデータモデルはシミュレーション内時間であるタイムスタンプと実際の作成日など独自の要素がいくつかある。共通化されたデータモデルを利用するためにはそれらを

変換, 翻訳する機能が必要となってくる.

# 謝辞

本論文を執筆するにあたり、多くの方からご助言とご助力をいただきました。主指導教員の丹康雄教授には、日頃のゼミや研究の議論、発表資料の作成の際にご丁寧な指導を受け賜りました。拙いゼミ発表や資料であったのにも関わらず、多くの知識や助言を頂き感謝しております。

、審査員を引き受けてくださったリム勇仁准教授、篠田陽一教授、BEURAN, Razvan Florin 准教授には中間発表などを通じて新たな視点から示唆に富んだアドバイスをいただきました。篠田陽一教授には中間発表後に研究についての意見交換をする場を設けてもらい感謝をしております。リム勇仁准教授には普段のゼミで毎回ご助言をいただき、研究を進める事ができました。

また、丹リム研究室の皆様には研究だけではなく、通信分野全般の知見を深める機会を頂きました。それだけではなく、日常生活まで手助けしていただき心から感謝しております。

最後に、学生生活を遠くから支えてくださった家族に感謝いたします。ありがとうございます。

## 参考文献

- [1] 鈴木 大章. 自然災害がもたらす人的被害軽減のためのデジタルツイン実現に関する調査研究. 2021.
- [2] David Jones, Chris Snider, Aydin Nassehi, Jason Yon, and Ben Hicks. Characterising the digital twin: A systematic literature review. *CIRP Journal of Manufacturing Science and Technology*, 29:36–52, 2020.
- [3] Eric J Tuegel, Anthony R Ingraffea, Thomas G Eason, and S Michael Spottswood. Reengineering aircraft structural life prediction using a digital twin. *International Journal of Aerospace Engineering*, 2011, 2011.
- [4] Mike Shafto, Mike Conroy, Rich Doyle, Ed Glaessgen, Chris Kemp, Jacqueline LeMoigne, and Lui Wang. Modeling, simulation, information technology & processing roadmap. *National Aeronautics and Space Administration*, 32(2012):1–38, 2012.
- [5] Michael Grieves. Digital twin: manufacturing excellence through virtual factory replication. *White paper*, 1(2014):1–7, 2014.
- [6] Aidan Fuller, Zhong Fan, Charles Day, and Chris Barlow. Digital twin: Enabling technologies, challenges and open research. *IEEE access*, 8:108952–108971, 2020.
- [7] National Research Foundation. Virtual singapore. <https://www.nrf.gov.sg/programmes/virtual-singapore> (2023/01/13).
- [8] 東京都 デジタルツインプロジェクト. <https://info.tokyo-digitaltwin.metro.tokyo.lg.jp/roadmap/> (2023/01/13).
- [9] 総務省. 情報通信白書. 2021.
- [10] 佐取 朗 石垣 武夫 藍 光郎, 室 英夫. 次世代センサハンドブック. 2008.
- [11] Somayya Madakam, Vihar Lake, Vihar Lake, Vihar Lake, et al. Internet of things (iot): A literature review. *Journal of Computer and Communications*, 3(05):164, 2015.

- [12] e-stat 政府統計ポータル. <https://www.e-stat.go.jp/> (2023/01/13).
- [13] 国立社会保障・人口問題研究所 将来推計人口・世帯数. <https://www.e-stat.go.jp/> (2023/01/13).
- [14] My city report. <https://www.mycityreport.jp/> (2023/01/13).
- [15] Gerhard Gröger and Lutz Plümer. Citygml–interoperable semantic 3d city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71:12–33, 2012.
- [16] Kristen Barlish and Kenneth Sullivan. How to measure the benefits of bim— a case study approach. *Automation in construction*, 24:149–159, 2012.
- [17] 鳥海不二夫, 山本仁志, et al. マルチエージェントシミュレーション: 1. マルチエージェントシミュレーションの基本設計. *情報処理*, 55(6):530–538, 2014.
- [18] Jagurs. <https://github.com/jagurs-admin/jagurs> (2023/01/13).
- [19] 青井真, 早川俊彦, and 藤原広行. 地震動シミュレータ: Gms. *物理探査 = BUTSURI-TANSA Geophysical Exploration*, 57(6):651–666, 2004.
- [20] Jia Liu, Yun Chen, and Yong Chen. Emergency and disaster management–crowd evacuation research. *Journal of Industrial Information Integration*, 21:100191, 2021.
- [21] Yoko Hasegawa, Yoshihide Sekimoto, Toshikazu Seto, Yuki Fukushima, and Midori Maeda. My city forecast: Urban planning communication tool for citizen with national open data. *Computers, Environment and Urban Systems*, 77:101255, 2019.
- [22] 慧 廣井, 朋哉 井上, 邦夫 明石, 颯 廣中, 洋信 菅野, 翼 湯村, 利幸 宮地, and 陽一 篠田. Aria : シミュレーション・エミュレーション連携基盤を利用したインタラクティブな都市型水害の被害予測システム. *情報処理学会論文誌コンシューマ・デバイス&システム (CDS)* , 10(2):11–25, 05 2020.
- [23] Mqtt. <https://mqtt.org/getting-started/> (2023/01/13).