Towards Self-Adaptive Data Management in Digital Twins for Biodiversity Monitoring

Eduard Kamburjan*†, Laura Ann Slaughter†‡, Einar Broch Johnsen†, Andrea Pferscher†, and Laura Weihl*
*IT University of Copenhagen, Denmark, Email: eduard.kamburjan@itu.dk
†University of Oslo, Norway,
‡SINTEF AS, Norway

***Abstract*—Biodiversity monitoring is concerned with keeping track of different species in an ecosystem over time, with respect to their abundance, distribution and diversity. Environmental digital twins used for biodiversity monitoring share characteristics with industrial digital twins, but face additional challenges in connecting data and models: Biodiversity data is often not livestreamed, interventions are slow and require human interaction, and the scientific knowledge about species and their habitats constantly evolves. Today, environmental digital twins offer little automation support, or any support to help scientists link species observations to assumptions about biodiversity. This paper presents an application of structural self-adaptation, originally developed in the industrial domain, to environmental digital twins. We show how structural self-adaptation enables to autonomously adapt monitored assumptions to changes in the available data sources, and further discuss how digital twins can adapt to changes in the domain knowledge. A first evaluation is given based on underwater cameras in the Oslo Fjord.**

***Index Terms*—Environmental Digital Twins, Biodiversity Monitoring, Self-Adaptation**

I. Introduction

At their core, digital twins (DTs) synchronize physical entities with their digital representations, mostly models, at an appropriate rate [1], to provide some services to users. DTs arose as a concept in the manufacturing industry but have spread far beyond their original domain: *environmental DTs* are used in environmental applications, in particular biodiversity monitoring, where observations of different species in nature are explored and explained through a multitude of physical, biological, and ecological models [2].

Consider a DT for an underwater habitat as a case in point. The DT can monitor assumptions or hypotheses over observations that link to ecological niche information for different species. Such assumptions can target general effects, such as population collapse due to anthropogenic nutrient pollution [3], or to the influence of offshore wind farms on biodiversity [4]. Another specific assumption is that tropical invasive fish species may occur in summer around major ports on the Norwegian coast, where they are disposed of by international ships, but do not survive harsh winters [5]. Any such monitored assumption must continuously adapt for two reasons.

This work is supported by the *Digital Arctic Twins* research network, funded by UArctic, and the *Digital Twin of the Oslo Fjord project* of the University of Oslo. We thank Anemo Robotics for making their species detection model, and Marinreparatørene for making their facilities available to us.

- The *occurrence of new data*, i.e. new observations, can be spurious. Designing and deploying cameras for long-term underwater monitoring is notoriously difficult, especially offshore, where these systems must operate without a direct power supply or access to maintenance. Observations arrive in batches from surveys [6], or irregularly from underwater cameras, as the battery life and hardware limitations of these cameras require periodic maintenance and retrieval cycles.
- The *knowledge which species are considered invasive*, or the description of the ecological niche of a species is richer/more accurate, or the behavior and tolerance levels of the species have adapted to new conditions [7], [8].

Environmental DTs face not only the challenges of industrial DTs, for example, management of heterogeneous models and data integration, but also have additional characteristics that hamper their wide-spread adaptation: Data sources are scarce, rarely in real-time (especially for underwater monitoring), and the models change far quicker than in an industrial setting, as they do not mirror the design of a system, but the ever-evolving scientific knowledge about species and their preferred habitats [6]. Current platforms focus on data integration, model management, and high-performance computing [2], [9], but lack automation support.

In biodiversity monitoring [10], the physical twin includes the set of species observations, together with spatial and temporal information. The DT will then consist of the monitored assumption and the corresponding internal reasoning structure, including ecological niche models (ENM), defining the ecological conditions within which a species can survive and reproduce [11]. The reasoning structure mirrors the assumption through runtime monitors that require a combination of incoming data-streams. Both available data sources (but not the observational data itself) and the reasoning network are available in a knowledge graph [12], where they can be combined with domain ontologies and enable the application of the above techniques.

The main contribution of this paper is the application of self-adaptation to environmental DTs, in particular we show that the challenge of autonomously adapting a monitored assumption can be addressed through techniques to adapt the DT according to consistency conditions on its representation as a knowledge graph [13]: *Structural self-adaptation* can be expressed in terms of consistency conditions over the knowledge graph [14]. Each such condition compares the reasoning structure with the observation data sources. For example, if the occurrence of a new data source is relevant for a reasoning network, the DT will automatically adapt the structure to reestablish *consistency*. On the other hand, if the knowledge about the domain changes then *declarative lifecycle management* can adapt the underlying consistency conditions [15]: Each set of consistency conditions is linked with a predicate over reasoning structures *and domain knowledge* that states when these conditions have to hold. Changing domain knowledge corresponds to the addition of new predicates. We present a preliminary evaluation based on the Oslo Fjord DT, using data from underwater cameras with a species detection model as the data processor.

II. Background

A. Environmental DTs

Environmental DTs differ from industrial twins [2], [6]; they are complex digital representations of natural environments and large-scale ecosystems, as opposed to models of a single, albeit composed, asset. Like all DTs, environmental twins require live data to ensure that the twin is continuously updated; the data for these natural environments is highly diverse, multi-modal and multi-scale, coming from a multitude of sources, ranging from satellite imaging to eDNA analyses.

Domain knowledge is the foundation of environmental DTs, either implicitly for management and integration of real-time observational data and simulation models [11], [16], or explicitly to drive simulation for scientific exploration. Data and information models enable integration and the ability to make use of observational data streaming from multiple sources.

Environmental DTs can include historical data and a knowledge base of known scientific facts surrounding the ecosystem. Such models drive simulations and predictive functionality in the twins, for example ocean circulation models [17]. Environmental twins form an ecosystem to serve stakeholders with a wide range of needs, such as policymaking, restoration activities, invasive species monitoring, but also scientific exploration of new hypotheses and assumptions [18].

Challenge: The reliance on knowledge models results, among other, in one major challenge: There are no methods to update the *usage* of information models and data. New data sources appear frequently, and scientific knowledge is evolving, yet updating long-running experiments or integrating new knowledge is so far manual. It is this challenge of *adapting to changes in data and knowledge* we address here.

B. Underwater Biodiversity Monitoring

Biodiversity monitoring in an underwater environment is a highly challenging domain, as it is characterized by logistical and technological constraints that limit data availability and quality. Traditional sampling approaches rely heavily on manual surveys conducted by scientific divers, fishing trawls or benthic grabs [19], but these methods are expensive, lack scalability, and are potentially destructive or disturbing marine life [20], rendering them unsuitable for sustained long-term monitoring efforts. Biodiversity monitoring is moving towards more non-invasive sensor-based approaches, like in situ camera setups [21], [22], remote sensing [23], passive acoustic sensors [24] or environmental DNA sniffers [25].

Despite these advances, live links to underwater monitoring systems are still rare since physical constraints of the underwater setting severely restrict wireless communication methods [26]. Setups for long-term deployment are limited by hardware constraints like battery life or storage capacity [20], [27]. For camera setups, regular maintenance is required also to overcome aspects like biofouling [28], i.e. marine growth on the camera lens, which degrades the image quality over time and complicates long-term unattended operation.

As a result, underwater monitoring observations often lack temporal resolution; data arrives intermittently rather than continuously [27]. In between these data collection events, there might be long gaps with no incoming information, which complicates data and model integration [6].

C. Self-Adaptation and Consistency in DTs

Integration of different models, their links to the physical entities and the management of their consistency are major challenges in DTs [29], [30], that led to the development of DT platforms [31] to support the user. While such platforms have different focuses and approaches, and we refer to the survey of Lehner et al. [31] for an overview, two technologies have arisen as tools for consistency management: knowledge graphs and self-adaptation.

Knowledge graphs and ontologies [12] are a key technology to integrate data within a DT, and reflect on the relations between models [32], [33]. They enable to automatically detect inconsistencies by analyzing the links between models [15], [34] or to connect to standards or data models.

Self-adaptation is the autonomous reaction of a system to unforeseen new situations or inputs [35]. Architectural [36] or structural [14] self-adaptation, which is explored in software systems beyond DTs [37], reacts to unforeseen *configurations* of the structure in the system, such as inconsistent linkage between the digital representations in the twin. Knowledge graphs have been shown to be a suitable foundation for DTs to autonomously react to structural inconsistency [14], [15].

III. Reasoning Networks

A DT manages the synchronization between physical (or *actual*) and digital entities to provide services through the digital entities. In our case, the physical entities are physical observations in nature, i.e., the data sources. This is akin to industrial DTs in, e.g., manufacturing, where the physical entities are not directly part of the DT, but accessible via digital references to sensors and actuators. The digital entities are the assumptions about observations and auxiliary services, in particular the data links to the data sources.

We think of the set of the physical entities as the *observational network*, and the monitored assumptions with auxiliary services as the *reasoning network*. The observational network contains both historical and real-time data sources. The reasoning network contains posed queries, i.e., biodiversity assumptions, to reason about the absence or presence of observations in the data sources. Each assumption is associated with has a set of auxiliary data link services that connect to the observational network and provide evidence for or against a query.

Consistency is intuitively defined by data links to exactly the relevant data sources, for each assumption. To express the observations that are relevant for a monitored assumption, we consider different kinds of conditions as constraints:

Definition 1 (Conditions):

- A *spatial condition* $\mathtt{sc}$ describes the location of an observation. In our case, it is a subset of $\mathbb{R}^2$ and models longitude and latitude.
- A *temporal condition* $\mathtt{tc}$ describes the time when an observation was made. In our case, it is a subset of $\mathbb{N}^+$ and models time in epoch.
- An *environmental condition* $\mathtt{ec}$ describes under which condition an observation was made. In our case, it is a first-order predicate over the environmental predicates.
- A *species condition* $\mathtt{oc}$ describes which species an observation describes. In our case, it is a subset of all taxons.

Note that temporal conditions need not be intervals and spatial conditions need not be connected regions; they may express, e.g., periods or disconnected unions of connected regions.

In the following, we leave the set of environmental conditions under-specified in this work, and use the GBIF taxons [38] as the species conditions.

Example 1: As a running example of an environmental DT, we consider the following assumption, derived from the literature [39]: There are no observations of cod (1) in the Oslo Fjord, (2) when the water is warmer than 18 degrees, (3) at daytime in 2025. The species condition $\mathtt{oc}_{\mathsf{cod}}$ is the GBIF taxon id of *Gadus morhua*: 8084280. The environmental condition $\mathtt{ec}_{\mathsf{cod}}$ is the expression $\mathit{temperature} \leq 18$ and the temporal condition $\mathtt{tc}_{\mathsf{cod}}$ is the set of all timestamps between sunrise and sunset of the days in the year 2025. Finally, the spatial condition $\mathtt{sc}_{\mathsf{cod}}$ is a rectangle covering the whole Oslo Fjord: $\{(x, y) \mid 59.25 \leq x \leq 60.00, 10.10 \leq y \leq 10.83\}$.

A. The Observational Network

An observational network is a set of data sources. Each data source in the observational network is either a historical or real-time. In either case, we model them as a time-stamped series of observations with its spatial position. Let V be the set of observations which are pairs of taxon ids and additional sensor measurements, e.g., temperature.

Definition 2 (Observational Network):

- An *observation* $v \in V$ is a pair $\langle \mathtt{taxon}, D\rangle$, where $\mathtt{taxon}$ is a GBIF taxon id and D a set of environmental measurements.
- A *data source* is a triple $\mathsf{src} = \langle \mathtt{dat}, \mathtt{pos}, \mathtt{id}\rangle$, where $\mathtt{dat} : \mathbb{R}^+ \rightharpoonup \mathtt{V}$ maps timestamps to observations, $\mathtt{pos} \in \mathbb{R}^2$ is a position, and $\mathtt{id} \in \mathbb{N}$ is an identifier.
- An *observational network* onet is a set of data sources. Given an index set I and an indexed set of data sources $\{\mathsf{src}_i | i \in I\}$, we write $\mathsf{onet} = \langle \mathsf{src}_i\rangle_I$.

We denote the i'th element of a tuple $t = \langle e_1, \dots\rangle$ with $e_i(t)$.

Observational networks evolve: new historical data sources are added and new observations are added through real-time datastreams. Formally, an observational network can evolve in two ways, which we model this as two different transitions.

We formalize evolution as a transition system, with two kinds of transition to cover both cases. The underlying transition system gives us a formal model of the DT that is used to prove correctness of self-adaptation procedures, i.e., that they indeed reestablish consistency.

We denote the last time for which an observation is available in a data source src, resp. an observational network, with

$$\mathtt{mtim}(\mathsf{src}) = \mathbf{max}\ \mathbf{dom}\ \mathtt{dat}(\mathsf{src})$$
$$\mathtt{mtim}(\langle \mathsf{src}_i\rangle_I) = \mathbf{max}_{i \in I}\ \mathtt{mtim}(\mathsf{src}_i)$$

Definition 3 (Observational Evolution): Let $\mathsf{onet} = \langle \mathsf{src}_i\rangle_I$ be an observational network.

- Adding a new data source src_j with $j \notin I, \mathtt{mtim}(\mathsf{src}_j) \leq \mathtt{mtim}(\mathsf{onet})$ to onet is denoted as follows.
$$\langle \mathsf{src}_i\rangle_I \xrightarrow{\mathsf{src}} \langle \mathsf{src}_i, \mathsf{src}_j\rangle_{I \cup \{j\}}$$
The condition on maximal time ensures that the new data source has no data yet from the perceived future.
- Adding a new observation to existing data sources within $t \in \mathbb{N}^+$ time units to onet is denoted as follows.
$$\langle \mathsf{src}_i\rangle_I \xrightarrow{t} \langle \mathsf{src}'_i\rangle_I$$
with the following constraints for all $i \in I$:
$$\mathtt{id}(\mathsf{src}_i) = \mathtt{id}(\mathsf{src}'_i) \wedge \mathtt{pos}(\mathsf{src}_i) = \mathtt{pos}(\mathsf{src}'_i) \quad (1)$$
$$\forall x \leq \mathtt{mtim}(\langle \mathsf{src}_i\rangle_I).\ x(\mathtt{dat}(\mathsf{src}'_i)) = x(\mathtt{dat}(\mathsf{src}_i)) \quad (2)$$
$$\mathtt{mtim}(\langle \mathsf{src}'_i\rangle_I) = t + \mathtt{mtim}(\langle \mathsf{src}_i\rangle_I) \quad (3)$$
This expresses that all ids and positions are preserved (Eq. 1), that all observations prior to the advance are preserved (Eq. 2) and that the time advance is the maximal time advance for the added observations (Eq. 3).

Example 2: We continue with Example 1. Consider two different locations that are used for monitoring, where the DT of the Oslo Fjord project recorded underwater videos: One at the most narrow point of the fjord, near Drøbak, at $\mathtt{pos}_{\mathsf{Drøbak}}(59.6658, 10.6120)$ and one at the northermost position of Nessoddtangen at $\mathtt{pos}_{\mathsf{Nessodd}} = (59.8694, 10.6551)$. Their representation, without observations, is as follows.

$$\mathsf{src}_{\mathsf{Drøbak}} = \langle \mathtt{dat}_{\mathsf{Drøbak}}, \mathtt{pos}_{\mathsf{Drøbak}}, 1\rangle$$
$$\mathsf{src}_{\mathsf{Nessodd}} = \langle \mathtt{dat}_{\mathsf{Nessodd}}, \mathtt{pos}_{\mathsf{Nessodd}}, 2\rangle$$

B. The Reasoning Network

An assumption over biodiversity observations is an expression over different time periods, locations and species, rooted in prior domain knowledge, and expressed by a domain expert. For our purposes, an assumption is a central component that we formalize as a *monitored assumption*, together with *data* *links* that connect assumption monitors to the observation network. Reasoning takes place in the assumption monitor that creates a verdict on an underlying assumption by analysing different data sources.

We do not model the internal structure of the assumption monitor, which we conjecture could consist of several nodes, but are interested in the decomposition of an assumption into an assumption monitor and data links. This decomposition is also the basis for consistency. Formally, we describe decomposition and consistency using *monitoring rules*, which describe which data links must exist for a given assumption.

Definition 4 (Monitoring Rule and Data Conditions):

- A *data condition* is a triple $\mathsf{dc} = \langle \mathsf{pol}, \mathsf{ec}, \mathsf{oc} \rangle$, where $\mathsf{pol} \in \{+, -\}$ denotes whether the assumption reacts to the presence $(+)$ or absence $(-)$ of data fitting the environmental condition ec and species condition oc.
- A *monitoring rule* is a tuple $\mathsf{rule} = \langle rid, \mathsf{sc}, \mathsf{tc}, \mathsf{DC} \rangle$, where rid is an identifier and DC a set of data conditions.

A monitoring rule describes the spatial sc and temporal tc regions relevant to an assumption, as well as the conditions on the observations to validate the underlying assumption.

Example 3: We continue with Example 2. The data conditions for our assumption are as follows. They are derived from the assumption. Note that we have two data conditions: the first confirms the assumption, while the second rejects it by filtering for counterexamples: cods that occur in water that is assumed to be too warm.

$$\mathsf{dc}^+_{\mathsf{cod}} = \langle +, \mathsf{ec}_{\mathsf{cod}}, \mathsf{oc}_{\mathsf{cod}} \rangle \qquad \mathsf{dc}^-_{\mathsf{cod}} = \langle -, \neg \mathsf{ec}_{\mathsf{cod}}, \mathsf{oc}_{\mathsf{cod}} \rangle$$

The corresponding rule is $\mathsf{rule}_{\mathsf{cod}}$:

$$\mathsf{rule}_{\mathsf{cod}} = \langle 3, \mathsf{sc}_{\mathsf{cod}}, \mathsf{tc}_{\mathsf{cod}}, \{\mathsf{dc}^+_{\mathsf{cod}}, \mathsf{dc}^-_{\mathsf{cod}}\} \rangle$$

Formally, reasoning networks consists of two kinds of elements: *Assumption monitors*, which result from queries and are confined by spatial conditions sc and temporal conditions tc. They are connected to the second kind of element, *data links*. A data link manage the connection between an assumption, one data source, and one data condition.

Definition 5 (Reasoning Network):

- An *assumption monitor* is a tuple $\mathsf{mon} = \langle aid, rid, \mathsf{sc}, \mathsf{tc} \rangle$, where aid is its identifier, rid is the id of the monitoring rule it uses, and sc and tc are spatial and temporal conditions.
- A *data link* is a tuple $\mathsf{link} = \langle lid, aid, id, \mathsf{dc} \rangle$, where lid is its identifier, aid is the identifier of the assumption monitor it is connected to, id is the identifier of the data source it is connected to, and dc the data condition it uses to filter observations.
- A *reasoning network* is a tuple $\mathsf{rnet} = \langle \mathsf{Mon}, \mathsf{Link} \rangle$, where Mon is a set of assumption monitors and Link is a set of data links, such that the aid of any $\mathsf{link} \in \mathsf{Link}$ is indeed the identifier of an assumption monitor $\mathsf{mon} \in \mathsf{Mon}$.

Example 4: We continue with Example 3. Consider the following reasoning network, that contains the assumption from Example 1 and data links that realize the connection to the data sources at Nessoddtangen.

$$\mathsf{rnet}^1_{\mathsf{cod}} = \Big\langle \big\{ \langle 4, 3, \mathsf{sc}_{\mathsf{cod}}, \mathsf{tc}_{\mathsf{cod}} \rangle \big\}, \big\{ \langle 7, 4, 2, \mathsf{dc}^-_{\mathsf{cod}} \rangle, \langle 6, 4, 2, \mathsf{dc}^+_{\mathsf{cod}} \rangle \big\} \Big\rangle$$

A reasoning network can also evolve: A user may add a new assumption to be monitored, which results in a new assumption monitor. It does *not* result in new data links. As we will see shortly, initial decomposition can be achieved using the very same mechanism as the one used to reestablish consistency.

Definition 6 (Reasoning Evolution): Let mon be an assumption monitor and $\mathsf{rnet} = \langle \mathsf{Mon}, \mathsf{Link} \rangle$ be a reasoning network with $\mathsf{mon} \notin \mathsf{Mon}$. Its addition is modelled as a transition

$$\langle \mathsf{Mon}, \mathsf{Link} \rangle \xrightarrow{\mathsf{mon}} \langle \mathsf{Mon} \cup \mathsf{mon}, \mathsf{Link} \rangle$$

Finally, a DT (for the scope of this work) is a triple of an observational network, a reasoning network, and a set of monitoring rules, such that all identifiers resolve.

Definition 7 (Digital Twins): A *digital twin* (DT) is a tuple

$$\mathsf{twin} = \langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \rangle$$

where onet is an observation network, rnet is a reasoning network, and Rule is a set of monitoring rules, s.t. the rid component of each assumption resolves to the identifier of a rule in Rule, and the id component of each data link resolves to a data source identifier of a data source in rnet.

DTs evolve if one of their components evolves (cf. Definitions 3 and 6), a new monitoring rule is added or the condition in a rule is modified. The last case corresponds to changes in the domain knowledge, such as different data conditions, or changed spatial/temporal regions in the monitored assumption. Evolution of DTs is defined in Figure 1.

Example 5: We continue with Example 4. The following is the state of the DT where only one data source is present, which then evolves by adding a new data source at Drøbak.

$$\begin{aligned} &\big\langle \{\mathsf{src}_{\mathsf{Nessodd}}\}, \mathsf{rnet}^1, \{\mathsf{rule}_{\mathsf{cod}}\} \big\rangle \\ \rightarrow &\big\langle \{\mathsf{src}_{\mathsf{Nessodd}}, \mathsf{src}_{\mathsf{Drøbak}}\}, \mathsf{rnet}^1, \{\mathsf{rule}_{\mathsf{cod}}\} \big\rangle \end{aligned}$$

The DT must now react to the new data source, which is obviously relevant for the assumption we monitor.

$$\begin{aligned} &\langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \rangle \rightarrow \langle \mathsf{onet}, \mathsf{rnet}', \mathsf{Rule} \rangle \\ &\qquad\qquad \text{if } \mathsf{rnet} \xrightarrow{\mathsf{mon}} \mathsf{rnet}' \text{ for some } \mathsf{mon} \\ &\langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \rangle \rightarrow \langle \mathsf{onet}', \mathsf{rnet}, \mathsf{Rule} \rangle \\ &\qquad\qquad \text{if } \mathsf{onet} \xrightarrow{\mathsf{src}} \mathsf{onet}' \text{ for some } \mathsf{src} \\ &\langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \rangle \rightarrow \langle \mathsf{onet}', \mathsf{rnet}, \mathsf{Rule} \rangle \\ &\qquad\qquad \text{if } \mathsf{onet} \xrightarrow{t} \mathsf{onet}' \text{ for some } t \\ &\langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \rangle \rightarrow \langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \cup \mathsf{rule} \rangle \\ &\qquad\qquad \text{if } \mathsf{rule} \notin \mathsf{Role} \\ &\langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \rangle \rightarrow \langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule} \setminus \{\mathsf{rule}\} \cup \{\mathsf{rule}'\} \rangle \\ &\qquad\qquad \text{if } rid(\mathsf{rule}) = rid(\mathsf{rule}') \end{aligned}$$

Fig. 1. Digital twin evolution

IV. Self-Adaptation

Consistency: Users use the DT to access data sources and monitor assumptions about observation. Returning the correct results from an assumption monitor requires the DT to be consistent: Each assumption monitor (a) must be linked to all relevant data sources through data links, and (b) these that monitors must use the data conditions according to the used monitoring rule. We define consistency formally in two steps: spatial consistency and rule consistency.

Spatial consistency states that for each data source in the spatial region of the assumption, a monitor exists with a data condition corresponding to the used monitoring rule.

Definition 8 (Spatial Consistency): Let $\mathsf{twin} = \langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule}\rangle$ be a DT, and $\mathsf{mon} \in \mathsf{Mon}(\mathsf{rnet})$ be an assumption monitor. We say that mon is *spatially consistent within* twin, if the following condition holds.

$$\forall \mathsf{src} \in \mathsf{onet}.\ \mathsf{pos}(\mathsf{src}) \in \mathsf{sc}(\mathsf{mon}) \rightarrow$$
$$\exists \mathsf{link} \in \mathsf{Link}(\mathsf{rnet}).\ aid(\mathsf{link}) = aid(\mathsf{mon}) \wedge id(\mathsf{link}) = id(\mathsf{src})$$

We say that twin is *spatially consistent*, if all assumption monitors within are spatially consistent within.

The second DT state in Example 5 is not spatially consistent. Spatial consistency states that the data link exists, while rule consistency states that the data links fit the rule underlying the assumption monitor.

Definition 9 (Rule Consistency): Let twin be a spatially consistent DT of the form $\langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule}\rangle$, and $\mathsf{mon} \in \mathsf{Mon}(\mathsf{rnet})$ an assumption monitor. Let $\mathsf{links}(\mathsf{mon})$ be the data links assigned to mon and $\mathsf{rule}(\mathsf{mon})$ its monitoring rule. We say that mon is *rule consistent in* twin, if the following holds.

$$\forall \mathsf{dcs} \in \mathsf{DC}\big(\mathsf{rule}(\mathsf{mon})\big).\ \exists \mathsf{link} \in \mathsf{links}(\mathsf{mon}).\ \mathsf{dc}(\mathsf{link}) = \mathsf{dcs}$$

We say that twin is *rule consistent*, if all assumption monitors within are rule consistent within. We say that a data link $\mathsf{link} \in \mathsf{links}(\mathsf{mon})$ *contributes* to mon, if it is a partial witness to its rule consistency, i.e., $\exists \mathsf{dcs} \in \mathsf{DC}\big(\mathsf{rule}(\mathsf{mon})\big).\ \mathsf{dc}(\mathsf{link}) = \mathsf{dcs}$.

Example 6: The first DT state in Example 5 is rule consistent. Removing any data link makes it rule inconsistent.

Repair: Consistency is declarative and an instance of *declarative lifecycle management* [15]. Each rule defines a membership predicate for elements that is considers (via the rid), and a consistency predicate for connected elements (via the consistency conditions above, which are *per assumption monitor*). Repair, is thus abduction: Given an assumption monitor which is *not consistent*, we can create a logic representation of the DT [13], [40] and use standard abduction reasoners to ask which elements must be added or removed to the logical representation to achieve consistency.

This is a general scheme, that can be expressed in the MAPE-K framework [35], [41] and does not require instantiating the repair function. To make the presentation in this work self-contained, we nonetheless instantiate it, but stress that this is not strictly necessary in general [40], [42].

Definition 10 (Consistent Evolution): Let $\mathsf{twin} = \langle \mathsf{onet}, \mathsf{rnet}, \mathsf{Rule}\rangle$ be a DT. The *repair function* $\mathtt{repair}(\mathsf{twin})$ that returns a rule consistent DT is given in Listing 1. It uses the following auxiliary functions.

```
function repair(Digital Twin twin)
  var links = Link(rnet(twin))
  //If the rule changed, remove all old links
  for mon ∈ norule(twin) do
    for link ∈ nolink(mon) do
      links := links \ {link}
    for src ∈ mis(mon), dc ∈ DC(rule(mon)) do
        links := links ∪ ⟨fresh(), aid(mon), id(src), dc⟩
  //For each source without correct links, create them
  for mon ∈ nost(twin, links) do
    for src ∈ mis(mon), dvc ∈ DC(rule(mon)) do
        links := links ∪ ⟨fresh(), aid(mon), id(src), dc⟩
  //Compute output
  var rnet := ⟨Mon(rnet(twin)), links⟩
  return ⟨onet(twin), rnet, Rule(twin)⟩
end
```

Listing 1. Repair Function

- Function $\mathsf{fresh}()$ creates a fresh data link identifier, i.e., an identifier not used anywhere in the system.
- Function $\mathsf{norule}(\mathsf{twin})$ returns all not rule-consistent assumption monitors.
- Function $\mathsf{nolink}(\mathsf{mon})$ returns all data links that are assigned to mon but do not contribute to its consistency.
- Function $\mathsf{nost}(\mathsf{twin}, \mathsf{links})$ returns spatially inconsistent monitored assumptions in twin, if the set links is used as the underlying set of data links.
- Function $\mathsf{mis}(\mathsf{mon})$ returns the set of data sources within the spatial region of mon that do not have a data link assigned to them and mon.

The *consistent evolution* of DTs is defined with a single transition $\Rightarrow$. Let $\Rightarrow^*$ be its reflexive-transitive closure.

$$\mathsf{twin} \Rightarrow \mathsf{twin}' \text{ if } \mathsf{twin} \rightarrow \mathsf{twin}'' \text{ and } \mathtt{repair}(\mathsf{twin}'') = \mathsf{twin}'$$

Our main formal result, which follows from the original framework on declarative self-adaptation [15] is that consistency can be ensured under evolution: After every step, the repair function reestablishes consistency. We consider running the repair function after the rarely occurring modifications of the structure justifiable. Note that the empty DT (i.e., no monitored assumptions, no data link, no data sources and no monitoring rules) is trivially rule consistent, so the condition to start in a rule consistent state is no restrictive in practice.

Theorem 1 (Consistency under Evolution): Let twin be a rule-consistent DT. Every DT twin' that is reachable from it, i.e., $\mathsf{twin} \Rightarrow^* \mathsf{twin}'$, is rule consistent.

Example 7: We continue with Example 4. The following is the evolution with self-adaptation into a rule consistent state, where the repaired reasoning network is $\mathsf{rnet}^2_{\mathsf{cod}}$.

$$\Big\langle \{\mathsf{src}_{\mathsf{Nesodd}}\}, \mathsf{rnet}^1, \{\mathsf{rule}_{\mathsf{cod}}\}\Big\rangle$$
$$\Rightarrow \Big\langle \{\mathsf{src}_{\mathsf{Nesodd}}, \mathsf{src}_{\mathsf{Drøbak}}\}, \mathsf{rnet}^2, \{\mathsf{rule}_{\mathsf{cod}}\}\Big\rangle$$
$$\text{with } \mathsf{rnet}^2_{\mathsf{cod}} = \Big\langle \Big\{ \langle 4, 3, \mathtt{sc}_{\mathsf{cod}}, \mathtt{tc}_{\mathsf{cod}}\rangle \Big\},$$
$$\Big\{ \langle 7, 4, 2, \mathtt{dc}^-_{\mathsf{cod}}\rangle, \langle 6, 4, 2, \mathtt{dc}^+_{\mathsf{cod}}\rangle, \langle 9, 4, 1, \mathtt{dc}^-_{\mathsf{cod}}\rangle, \langle 8, 4, 1, \mathtt{dc}^+_{\mathsf{cod}}\rangle \Big\} \Big\rangle$$

V. Prototype and Discussion

We have implemented our approach and applied it to observations from underwater cameras in the Oslo Fjord.[1] Our prototype uses an industrial computer vision model, described below. The model we provide is trained to detect Goldsinny wrasses (*Ctenolabrus rupestris*, GBIF taxon id 2383163, *Bergnebb* in Norwegian).

Implementation: The Oslo Fjord DT is an on-going interdisciplinary project to develop a framework able to answer user queries about the effect of climate and human stressors on marine habitats in the Oslo Fjord, by combining knowledge graphs, simulation models and runtime monitors. The framework combines sensor data (e.g., temperature and turbidity) with models capturing physical as well as biological aspects of the fjord system. In this context, we are here interested in complementing physical measurements with observations of marine life. To this aim, we deploy underwater cameras for shorter periods of time, restricted by battery capacity. These cameras are not able to transmit video live to the twin; instead, the recordings need to be collected manually, resulting in batched observations that are fed into a species detection model. Here, these video recordings constitute our data sources, while assumptions and links are added manually.

Our implementation is a proof-of-concept to explore self-adaptation to handle data sources in scientific exploration of underwater biodiversity observations, and is not yet fully integrated in the Oslo Fjord DT. The system consists of (1) a set of lightweight clients managing single data sources, realizing the observational network, (2) a server, which hosts the reasoning network and maintains the knowledge graph, and (3) a ActiveMQ instance serving as the middleware.

Each client manages one data source and maintains a database that stores the actual observations. The data sources are based on videos recorded in the Oslo Fjord. To extract observations, we use a species detection model provided by Anemic Robotics that generates a list of detected fish species using computer vision. We store its output, i.e., the label of the detected fish and the time of the recording, so each video is processed only once. An example detection is shown in fig. 2.

Each client is connected to a retroactive ActiveMQ queue and every time a listener registers on the queue, all observations are resend. This design is chosen to later enable the use of different models for detection on the same video. Each observation is sent via one message. Whenever a new data source is added, a new client is started and a special registration message is sent. The server implements the self-adaptation. Each time a data link is created, it connects to the corresponding ActiveMQ queue, filters according to the condition and streams the remaining messages to the assumption monitor. The monitor is added by the user, and outputs how many observations confirm or refute it.

Expected Benefits: Data Integration and Exploration: There are numerous EU platforms for sharing environmental twin data and models, such as The European Digital Twin of the Ocean, EDITO [43], the European Marine Observation and Data network, EMODnet [44], and others. For marine twins such as our Oslo Fjord DT, the EU Digital Twin of the Ocean (DTO), EDITO-infra data lake and EDITO Model Lab are relevant sharing access points for observations, historical data, and models. Standard organizations, such as the Open Geospatial Consortium [45], or Ocean Biodiversity Information System [46] are defining how data will be integrated for marine DTs. In this context, we expect that the proposed framework here will simplify the adaptive integration of these data sources by demonstrating a general method to connect them to on-going scientific explorations. In particular, we emphasize that we do not provide uniform data access, but uniform integration into self-adaptive layer, based on the ubiquitous GBIF data model for the critical conditions.

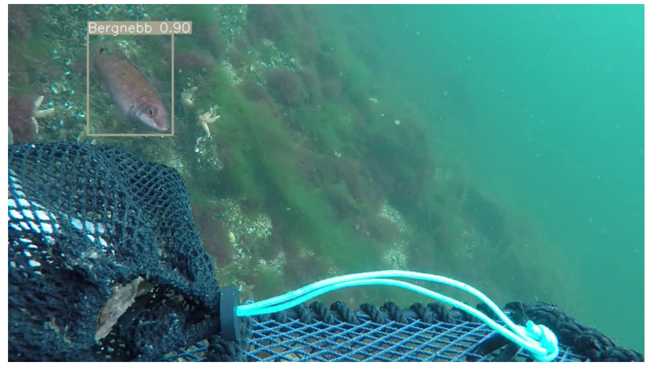

Fig. 2. Frame with one detected Goldsinny wrasse at Nesoddtangen.

Expected Benefits: Model Integration and Reasoning: We expect that the self-adaptation will increase the usability of the Oslo Fjord DT by providing automation support for data integration. Furthermore, we expect that our approach will enable easier model integration: While data links described here only connect a monitored assumption to a data source, they will be extended to also connect with simulation models to generate new data, change filters on the data source or employ different species detection models on stored data streams. We expect that such a connection will drastically reduce the need for manual model integration.

VI. Conclusion

DTs support management and integration of models and data, and face similar challenges for both engineered and natural systems. The synchronization between physical and digital entities, however, has to address different problems, due to the different ways data is collected and transmitted. This work shows that structural self-adaptation, first developed for DTs in engineering, can be applied to tackle the problems specific to underwater biodiversity monitoring, namely the delay of data acquisition and the complex structure domain experts have to navigate to monitor their assumptions. On the level of the formalization, an important aspect to investigate is that data and knowledge may both carry uncertainty, which must be handled by the network as well.

[1] For the implementation, see https://github.com/Edkamb/TwinExplore

References

[1] "ISO 23247-1, automation systems and integration–digital twin framework for manufacturing–part 1: Overview and general principles," International Organization for Standardization (ISO), Standard, 2021.

[2] K. de Koning, J. Broekhuijsen, I. Kühn, O. Ovaskainen, F. Taubert, D. Endresen, D. Schigel, and V. Grimm, "Digital twins: dynamic model–data fusion for ecology," *Trends in Ecology & Evolution*, vol. 38, no. 10, 2023.

[3] A. Orio, Y. Heimbrand, and K. Limburg, "Deoxygenation impacts on baltic sea cod: Dramatic declines in ecosystem services of an iconic keystone predator," *Ambio*, vol. 51, no. 3, Mar 2022.

[4] C. Li, J. W. P. Coolen, L. Scherer, J. M. Mogollón, U. Braeckman, J. Vanaverbeke, A. Tukker, and B. Steubing, "Offshore wind energy and marine biodiversity in the north sea: Life cycle impact assessment for benthic communities," *Environ. Sci. Technol.*, vol. 57, no. 16, 2023.

[5] V. Husa, A.-L. Agnalt, H. Berntsen, T. Falkenhaug, F. Fossøy, E. Forsgren, E. S. Grefsrud, A. M. Hjelset, F. Hansen, E. Husby, A. Jelmert, S. Mortensen, S. A. Olsen, and H. Sandvik, "Alien marine species in Norway: Mapping, monitoring and assessment of vectors for introductions," Norwegian Environment Agency, Tech. Rep., Apr. 2022.

[6] A. Trantas, R. Plug, P. Pileggi, and E. Lazovik, "Digital twin challenges in biodiversity modelling," *Ecol. Informatics*, vol. 78, 2023.

[7] A. Siwertsson, U. Lindstrøm, M. Aune, and R. Primicerio, "Rapid climate-driven shifts in coastal fish communities along Norway's coast: Evidence from 25 years of monitoring," *Global Change Biology*, 2024.

[8] C. A. Froyd, H. Jensen, A. van Leeuwen *et al.*, "Climate-induced habitat suitability modelling for pelagic fish in european seas," *Frontiers in Marine Science*, vol. 12, 2025.

[9] A. Agrawal, R. Thiel, P. Jain, V. Singh, and M. Fischer, "Digital twin: Where do humans fit in?" *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2301.03040

[10] A. Allard, E. C. H. Keskitalo, and A. Brown, Eds., *Monitoring Biodiversity: Combining Environmental and Social Data*. Routledge, 2023.

[11] S. M. Melo-Merino, H. Reyes-Bonilla, and A. Lira-Noriega, "Ecological niche models and species distribution models in marine environments: A literature review and spatial analysis of evidence," *Ecological Modelling*, vol. 415, 2020.

[12] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. F. Sequeda, S. Staab, and A. Zimmermann, "Knowledge graphs," *ACM Comput. Surv.*, vol. 54, no. 4, 2022.

[13] E. Kamburjan, A. Pferscher, R. Schlatte, R. Sieve, S. L. Tapia Tarifa, and E. B. Johnsen, "Semantic reflection and digital twins: A comprehensive overview," in *The Combined Power of Research, Education, and Dissemination*, ser. LNCS, vol. 15240. Springer, 2025.

[14] E. Kamburjan, V. N. Klungre, R. Schlatte, S. L. Tapia Tarifa, D. B. Cameron, and E. B. Johnsen, "Digital twin reconfiguration using asset models," in *ISoLA (4)*, ser. LNCS, vol. 13704. Springer, 2022.

[15] E. Kamburjan, N. Bencomo, S. L. Tapia Tarifa, and E. B. Johnsen, "Declarative lifecycle management in digital twins," in *EDTConf*. ACM, 2024.

[16] D. O. Ortiz-Rodríguez, A. Guisan, R. Holderegger, and M. J. van Strien, "Predicting species occurrences with habitat network models," *Ecology and Evolution*, vol. 9, no. 18, 2019.

[17] A. Tzachor, O. Hendel, and C. E. Richards, "Digital twins: a stepping stone to achieve ocean sustainability?" *npj Ocean Sustainability*, vol. 2, no. 1, Oct. 2023.

[18] U. Brönner, M. Sonnewald, and M. Visbeck, "Digital Twins of the Ocean can foster a sustainable blue economy in a protected marine environment," *The International Hydrographic Review*, vol. 29, 2023.

[19] M. Solan, J. D. Germano, D. C. Rhoads, C. Smith, E. Michaud *et al.*, "Towards a greater understanding of pattern, scale and process in marine benthic systems: a picture is worth a thousand worms," *Journal of experimental marine biology and ecology*, vol. 285, 2003.

[20] A. W. Bicknell, B. J. Godley, E. V. Sheehan, S. C. Votier, and M. J. Witt, "Camera technology for monitoring marine biodiversity and human impact," *Frontiers in Ecology and the Environment*, vol. 14, no. 8, 2016.

[21] M. Pedersen, J. Bruslund Haurum, R. Gade, and T. B. Moeslund, "Detection of marine animals in a new underwater dataset with varying visibility," in *CVPR*. CFV, 2019.

[22] R. Steenweg, M. Hebblewhite, R. Kays, J. Ahumada *et al.*, "Scaling-up camera traps: Monitoring the planet's biodiversity with networks of remote sensors," *Frontiers in Ecology and the Environment*, vol. 15, no. 1, 2017.

[23] D. Kachelriess, M. Wegmann, M. Gollock, and N. Pettorelli, "The application of remote sensing for marine protected area management," *Ecological Indicators*, vol. 36, 2014.

[24] C. Desjonquères, T. Gifford, and S. Linke, "Passive acoustic monitoring as a potential tool to survey animal and ecosystem processes in freshwater environments," *Freshwater Biology*, vol. 65, no. 1, 2020.

[25] A. D. Foote, P. F. Thomsen, S. Sveegaard, M. Wahlberg, J. Kielgast, L. A. Kyhn, A. B. Salling, A. Galatius, L. Orlando, and M. T. P. Gilbert, "Investigating the potential use of environmental dna (edna) for genetic monitoring of marine mammals," 2012.

[26] T. Theocharidis and E. Kavallieratou, "Underwater communication technologies: A review," *Telecommunication Systems*, vol. 88, no. 2, 2025.

[27] R. Danovaro, L. Carugati, M. Berzano, A. E. Cahill, S. Carvalho, A. Chenuil, C. Corinaldesi, S. Cristina, R. David *et al.*, "Implementing and innovating marine monitoring approaches for assessing marine environmental status," *Frontiers in Marine Science*, vol. 3, 2016.

[28] S. Zhang, J. Xi, J. Wu, P. Wang, F. Lin, and D. Zhang, "Design of an efficient antifouling strategy for underwater optical window based on chlorine generation," *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, vol. 634, 2022.

[29] M. van den Brand, L. Cleophas, R. Gunasekaran, B. Haverkort, D. Negrin, and H. Muctadir, "Models meet data: Challenges to create virtual entities for digital twins," in *MoDDiT@Models*. IEEE, 2021.

[30] F. Bordeleau, B. Combemale, R. Eramo, M. van den Brand, and M. Wimmer, "Towards model-driven digital twin engineering: Current opportunities and future challenges," in *Systems Modelling and Management*, Ö. Babur, J. Denil, and B. Vogel-Heuser, Eds. Springer International Publishing, 2020.

[31] D. Lehner, J. Pfeiffer, E. Tinsel, M. M. Strljic, S. Sint, M. Vierhauser, A. Wortmann, and M. Wimmer, "Digital twin platforms: Requirements, capabilities, and future prospects," *IEEE Softw.*, vol. 39, no. 2, 2022.

[32] E. Karabulut, S. F. Pileggi, P. Groth, and V. Degeler, "Ontologies in digital twins: A systematic literature review," *Future Gener. Comput. Syst.*, vol. 153, 2024.

[33] X. Zheng, J. Lu, and D. Kiritsis, "The emergence of cognitive digital twin: vision, challenges and opportunities," *Int. J. Prod. Res.*, vol. 60, no. 24, 2022.

[34] H. M. Muctadir, L. Cleophas, and M. van den Brand, "Maintaining consistency of digital twin models: Exploring the potential of graph-based approaches," in *SEAA*, 2024.

[35] D. Weyns, *An introduction to self-adaptive systems: A contemporary software engineering perspective*. John Wiley & Sons, 2020.

[36] V. A. Braberman, N. D'Ippolito, J. Kramer, D. Sykes, and S. Uchitel, "MORPH: a reference architecture for configuration and behaviour self-adaptation," in *CTSE@FSE*. ACM, 2015.

[37] E. Alberts, I. Gerostathopoulos, I. Malavolta, C. H. Corbato, and P. Lago, "Software architecture-based self-adaptation in robotics," *J. Syst. Softw.*, vol. 219, 2025.

[38] GBIF Consortium, "Global biodiversity information facility," 2025, https://www.gbif.org accessed 05.07.25.

[39] R. Froese and D. Pauly, *FishBase*, 2024, World Wide Web electronic publication. Version 06/2024. https://www.fishbase.org.

[40] E. Kamburjan, V. N. Klungre, R. Schlatte, E. B. Johnsen, and M. Giese, "Programming and debugging with semantically lifted states," in *ESWC*, ser. LNCS, vol. 12731. Springer, 2021.

[41] Y. Brun, G. D. M. Serugendo, C. Gacek, H. Giese, H. M. Kienle, M. Litoiu, H. A. Müller, M. Pezzè, and M. Shaw, "Engineering self-adaptive systems through feedback loops," in *Software Engineering for Self-Adaptive Systems*, ser. LNCS, vol. 5525. Springer, 2009.

[42] S. Gil, E. Kamburjan, P. Talasila, and P. G. Larsen, "An architecture for coupled digital twins with semantic lifting," *Software and Systems Modeling*, 2024.

[43] EDITO consortium, "European digital twin ocean," 2025, https://www.edito.eu/ accessed 05.07.25.

[44] EMODnet consortium, "European marine observation and data network," 2025, https://emodnet.ec.europa.eu accessed 05.07.25.

[45] OGC, "Open geospatial consortium," 2025, https://www.ogc.org accessed 05.07.25.

[46] OBIS Consortium, "Ocean biodiversity information syste," 2025, https://www.obis.org accessed 05.07.25.