ОРИГИНАЛЬНАЯ СТАТЬЯ
УДК 004.62
https://doi.org/10.26907/2541-7746.2025.2.367-383

Верификация интеграции данных в интегрированной системе баз данных по свойствам неорганических веществ и материалов

С.А. Ступников

Федеральный исследовательский центр «Информатика и управление» Российской академии наук, г. Москва, Россия

sstupniukov@ipiran.ru

Аннотация

С ростом неоднородности моделей и схем данных в современном мире все более необходимой становится интеграция данных. Системы интеграции данных создаются в различных предметных областях, например, в астрономии, управлении землепользованием и материаловедении. Программы интеграции данных могут быть очень сложными, а потому становятся важными вопросы формальной верификации их корректности.

В настоящей работе рассмотрен подход к верификации корректности интеграции данных в интегрированной системе баз данных по свойствам неорганических веществ и материалов Института металлургии и материаловедения им. А.А. Байкова РАН. Интеграция данных в этой системе проводится в два этапа: на первом этапе данные из источников, помеченные на удаление, изменение или добавление, преобразуются в промежуточное XML-представление; на втором этапе для элементов XML-представления вызываются процедуры целевой интегрированной базы данных, удаляющие, изменяющие или добавляющие в нее соответствующие записи. Реализация программ интеграции данных осуществлена с использованием композиции императивного языка программирования и декларативного языка реляционных баз данных. Подход к верификации основан на определении семантики схем данных и программ интеграции данных в формальном языке спецификаций и последующем доказательстве корректности интеграции данных с использованием автоматизированных средств доказательства.

Ключевые слова: интеграция данных, верификация, семантика программ, доказательство корректности

Благодарности. Работа выполнена в рамках темы государственного задания ФИЦ ИУ РАН. Автор благодарит Виктора Дударева (Ruhr University Bochum, ИМЕТ РАН) за ценные замечания к работе.

Для цитирования: *Ступников С.А.* Верификация интеграции данных в интегрированной системе баз данных по свойствам неорганических веществ и материалов // Учен. зап. Казан. ун-та. Сер. Физ.-матем. науки. 2025. Т. 167, кн. 2. С. 367–383.
https://doi.org/10.26907/2541-7746.2025.2.367-383.

ORIGINAL ARTICLE
https://doi.org/10.26907/2541-7746.2025.2.367-383

Verification of data integration in an integrated system of databases on the properties of inorganic substances and materials

S.A. Stupnikov

Federal Research Center "Computer Science and Control", Russian Academy of Sciences, Moscow, Russia

sstupniukov@ipiran.ru

Abstract

Due to the increasing heterogeneity of data models and schemas in the modern world, robust data integration is a high-priority issue. Data integration systems have been extensively deployed across various domains, including astronomy, land use management, and materials science. However, data integration programs can be very complicated. Thus, formal verification of their correctness has emerged as an important task.

In this article, an approach to verify the correctness of data integration in an integrated system of databases on the properties of inorganic substances and materials is considered. The system, developed at the A.A. Baikov Institute of Metallurgy and Materials Science of the Russian Academy of Sciences, employs a two-stage data integration process: during the first stage, the source data marked for deletion, modification, or insertion are converted into an intermediate XML representation; in the second stage, the system invokes the corresponding procedures for XML elements in the target integrated database and updates it accordingly. The data integration programs are implemented by combining an imperative programming language with a declarative language of relational databases. Verification is performed by defining the semantics of the data schemas and data integration programs in a formal specification language and proving the correctness of data integration using automated provers.

Keywords: data integration, verification, program semantics, proof of correctness

Acknowledgments. This study was carried out as part of the state assignment to the Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences. Sincere thanks are due to Victor Dudarev (Ruhr University Bochum, A.A. Baikov Institute of Metallurgy and Materials Science of the Russian Academy of Sciences) for his valuable comments.

For citation: Stupnikov S.A. Verification of data integration in an integrated system of databases on the properties of inorganic substances and materials. *Uchenye Zapiski Kazanskogo Universiteta. Seriya Fiziko-Matematicheskie Nauki*, 2025, vol. 167, no. 2, pp. 367–383.
https://doi.org/10.26907/2541-7746.2025.2.367-383. (In Russian)

Введение

В современном мире в науке и промышленности быстро растет число неоднородных источников данных. Каждый такой источник разработан на основе удобных его пользователям модели данных и системы управления базами данных (СУБД), а схема базы данных определена из соображений конкретной предметной области. Для преодоления неоднородности моделей и схем данных системы интеграции данных [1] создаются в различных предметных областях, например, в астрономии [2], управлении землепользованием [3] и материаловедении [4].

Программы интеграции данных в разрабатываемых системах могут быть очень сложными, а потому возникают вопросы формальной верификации корректности интеграции данных. Формальная верификация программ также достаточно сложна, однако ее применение при разработке программных систем оправдано, поскольку стоимость исправления ошибки после выпуска системы в производство может превышать стоимость исправления ошибки на этапе разработки системы в десятки и сотни раз [5].

Вопросы верификации интеграции данных исследуются достаточно активно. Обычно основная идея подходов к верификации состоит в том, чтобы сообщить программам интеграции данных семантику в некотором формальном языке. Свойства программ, подлежащие проверке, представляются в виде выражений этого языка. Затем с использованием формальных средств доказательства спецификация, выражающая семантику конкретной программы интеграции данных, проверяется на соответствие необходимым свойствам. Например, известны работы по определению формальной семантики самой распространенной в мире модели данных – языка SQL (например, [6]). Отдельное крупное направление образуют работы по верификации трансформаций моделей, основанной на движимой моделями инженерии (MDE) [7].

В направлении собственно интеграции данных в качестве языка определения формальной семантики моделей данных хорошо зарекомендовала себя «Нотация абстрактных машин» (AMN) [8] – язык, основанный на логике первого порядка и теории множеств. Язык AMN поддержан промышленным инструментарием, направленным на автоматизированную проверку корректности спецификаций Atelier [9]. Накоплен более чем двадцатилетний мировой опыт применения языка AMN и средств его инструментальной поддержки при разработке промышленных программных систем [10]. Методы верификации интеграции данных с использованием AMN предложены, в частности, в [11] и [12]. В работе [11] определена формальная семантика языка разрешения сущностей и слияния данных HIL в языке AMN для верификации потоков работ интеграции данных. В [12] разработан метод верификации корректности виртуальной интеграции данных в модели RDF. Корректность доказывается путем отображения схем предметной области, схем источников данных и запросов в язык AMN и последующего применения автоматизированных средств доказательства.

Настоящая работа имеет практическую направленность – верификацию интеграции данных в интегрированной базе данных Института металлургии и материаловедения (ИМЕТ) им. А.А. Байкова РАН [13], [14]. Рассмотрена интеграция данных из базы данных Bandgap [15], содержащей данные о ширине *запрещенной зоны* основных классов неорганических веществ. «Ширина запрещенной зоны является фундаментальным параметром конденсированных фаз, который характеризует природу химической связи в материале. По величине запрещенной зоны можно судить о типе химической связи, доминирующей в соединении, устойчивости соединения в определенном интервале изменений состава и

внешних параметров, типе электронной проводимости в образцах, склонности материала к ионной проводимости, а также основных термодинамических характеристиках соединения»[1].

Интеграция данных в системе проводится в два этапа: на первом этапе данные из базы данных Bandgap, помеченные на удаление, изменение или добавление, преобразуются в промежуточное XML-представление; на втором этапе для элементов XML-представления вызываются процедуры целевой интегрированной базы данных, удаляющие, изменяющие или добавляющие в нее соответствующие записи. Реализация программ интеграции данных проводится с использованием композиции императивного языка программирования (VBScript) и декларативного языка реляционных баз данных (SQL).

Статья организована следующим образом. В разделе 1 приведены структура схем данных и программ интеграции данных в интегрированной базе данных ИМЕТ и общая структура спецификаций языка AMN, выражающие формальную семантику программ интеграции данных. Определение формальной семантики схемы источника данных, схемы промежуточного представления данных, схемы интегрированной базы данных, программы извлечения данных из источников и преобразования в промежуточное представление, процедуры загрузки (обновления) данных в интегрированную базу данных продемонстрировано на примерах интеграции базы данных Bandgap. В разделе 2 проиллюстрирована верификация программ интеграции данных.

1. Формальная семантика программ интеграции данных

Структура схем данных и программ интеграции данных в интегрированной базе данных ИМЕТ РАН [14] включает следующие компоненты:

- реляционные схемы источников данных;
- схема промежуточного представления данных на языке XML Schema;
- программы извлечения данных из источников и преобразования в промежуточное XML-представление (композиция императивного языка программирования VBScript и декларативного языка реляционных баз данных SQL);
- реляционная схема интегрированной базы данных;
- процедуры загрузки (обновления) данных в интегрированную базу данных.

Программы интеграции данных запускают с установленной периодичностью (обычно ежедневно), извлекают накопившиеся изменения в источниках данных, формируют файлы промежуточного XML-представления, а затем для каждого из элементов XML-файлов запускают процедуры обновления данных в интегрированной базе данных.

Определение формальной семантики схем данных и программ интеграции данных, а также их дальнейшая верификация осуществляются для каждого источника данных независимо. Структура семантических спецификаций для отдельного источника данных включает следующие компоненты:

- AMN-спецификация вида MACHINE [8], определяющая семантику реляционной схемы источника данных и операций извлечения данных из источника;

[1] https://bg.imet-db.ru/default.asp?lang=ru.

- AMN-спецификация вида MACHINE, определяющая семантику схемы промежуточного XML-представления и операций создания элементов XML-документа;
- AMN-спецификация вида MACHINE, определяющая семантику реляционной схемы интегрированной базы данных и процедур обновления данных в интегрированной базе данных;
- AMN-спецификация вида REFINEMENT [8], определяющая семантику императивной программы интеграции данных, последовательно вызывающей операции извлечения данных из источника, преобразования данных в промежуточное представление, обновления данных в интегрированной базе данных.

Ниже в данном разделе формальная семантика программ интеграции данных проиллюстрирована на примере базы данных Bandgap. Все упомянутые файлы схем, программ и AMN-спецификаций опубликованы в репозитории GitHub[2].

1.1. Семантика реляционной схемы источника данных и операций извлечения данных из источника. Фрагмент реляционной схемы базы данных Bandgap приведен в файле *Bandgap-ver.sql*. Пример определения таблицы свойств веществ выглядит следующим образом[3]:

```
CREATE DATABASE [BandGap]
CREATE TABLE [dbo].[_PropertiesConv](
        [NOMPROP] [int] NOT NULL,
        [UpdateStatus] [int] NOT NULL,
        [NAZVPROP] [varchar](128) NOT NULL,
        [HTML] [varchar](128) NOT NULL,
 CONSTRAINT [PK___PropertiesConv] PRIMARY KEY ([NOMPROP])
)
```

Соответствующий фрагмент AMN-спецификации `Bandgap`, выражающий семантику реляционной схемы, выглядит следующим образом:

```
MACHINE  Bandgap
DEFINITIONS
  PropertiesConv_struct == struct(
       NOMPROP: INT,
       UpdateStatus: INT,
       NAZVPROP: seq(0..255),
       HTML: seq(0..255)
  );
ABSTRACT_VARIABLES
  PropertiesConv
INVARIANT
```

[2]https://github.com/sstupnikov/DataTransformation/tree/main/Bandgap.

[3]Строго говоря, рассматриваемые таблицы не являются непосредственно источником данных. Их наполнение происходит за счет использования триггеров на таблицах исходной базы данных. В настоящей работе этот аспект не рассмотрен для сокращения изложения, поскольку он добавляет не методологическую, а техническую сложность.

```
  PropertiesConv: FIN(PropertiesConv_struct) &
    !(prop1, prop2).(
        prop1: PropertiesConv & prop2: PropertiesConv &
        prop1'NOMPROP = prop2'NOMPROP =>
        prop1 = prop2)
INITIALISATION
  PropertiesConv:= {}
```

Определяются структура данных `PropertiesConv_struct` и переменная `PropertiesConv`, выражающие семантику таблицы `_PropertiesConv`. Переменная типизируется в разделе INVARIANT (ограничение первичного ключа выражается соответствующей формулой) и инициализируется пустым множеством.

Семантика SQL-операции извлечения обновленных данных из источника

```
SELECT * FROM _PropertiesConv WHERE UpdateStatus>0
```

выражается операцией машины `Bandgap`:

```
OPERATIONS
  result <-- select_PropertiesConv =
  result := { rcrd | rcrd: PropertiesConv & rcrd'UpdateStatus > 0 }
END
```

1.2. Семантика схемы промежуточного XML-представления и операций создания элементов XML-документа. Фрагмент схемы промежуточного представления приведен в файле *MUService-ver.xsd*. Пример определения XML-элемента `PropertiesInfo`, задающего свойства вещества, выглядит следующим образом:

```
<xs:schema targetNamespace="http://meta.imet-db.ru/MUService.xsd"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="MetaBase" type="MetaBaseType" />
<xs:complexType name="MetaBaseType">
 <xs:sequence>
  <xs:element name="PropertiesInfo" minOccurs="0" maxOccurs="1">
   <xs:complexType> <xs:sequence>
    <xs:element name="PropertiesInfoItem" minOccurs="1">
     <xs:complexType>  <xs:sequence>
       <xs:element name="PropID" type="xs:integer"
          minOccurs="1" maxOccurs="1" />
       <xs:element name="Name" type="xs:string"
          minOccurs="1" maxOccurs="1" />
       <xs:element name="Description" type="xs:string"
          minOccurs="1" maxOccurs="1" />
       <xs:element name="WWWTemplatePage" type="xs:string"
          minOccurs="1" maxOccurs="1" />
       <xs:element name="UpdateStatus" type="xs:integer"
          minOccurs="1" maxOccurs="1" />
      </xs:sequence> </xs:complexType>
  </xs:element>
```

```
  </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

Соответствующий фрагмент AMN-спецификации `MUService`, выражающей семантику XSD-схемы, выглядит следующим образом:

```
MACHINE  MUService
DEFINITIONS
    PropertiesInfoItem == struct(
        PropID: INT,
        Name: seq(0..255),
        Description: seq(0..255),
        WWWTemplatePage: seq(0..255),
        UpdateStatus: INT
    );
ABSTRACT_VARIABLES
    MetaBase
INVARIANT
    MetaBase: struct(
       PropertiesInfo: FIN(PropertiesInfoItem)
    )
```

Определяются структура данных `PropertiesInfoItem`, выражающая семантику одноименного XML-элемента, и переменная `MetaBase`, также выражающая семантику одноименного XML-элемента. Переменная типизируется в разделе INVARIANT. Можно видеть, что конструктор типов `complexType` представляется в AMN конструкцией структуры `struct`, а конструктор `sequence` – типизацией элемента множества конечных подмножеств FIN.

Семантика операции `createNode` (заданной для объектов типа `Msxml2.DOMDocument`) создания элемента `PropertiesInfoItem` выражается соответствующей операцией машины `MUService`:

```
createNodePropertiesInfoItem(value) =
PRE value: PropertiesInfoItem
THEN
  MetaBase'PropertiesInfo := MetaBase'PropertiesInfo \/ {value}
END;
```

1.3. Семантика реляционной схемы интегрированной базы данных и процедур обновления данных в интегрированной базе данных. Фрагмент реляционной схемы интегрированной базы данных Metabase приведен в файле *Metabase-ver.sql*. Пример определения таблицы свойств веществ выглядит следующим образом:

```
CREATE DATABASE [Metabase]
CREATE TABLE [dbo].[PropertiesInfo](
        [DBID] [int] NOT NULL,
        [PropID] [int] NOT NULL,
```

```
        [Name] [varchar](256) NOT NULL,
        [Description] [text] NOT NULL,
        [WWWTemplatePage] [varchar](256) NOT NULL,
        [UpdateStatus] [int] NOT NULL,
 CONSTRAINT [PK_PropertiesInfo] PRIMARY KEY ([DBID],[PropID])
)
```

Соответствующий фрагмент AMN-спецификации `Metabase`, выражающий семантику реляционной схемы интегрированной базы данных, выглядит следующим образом:

```
MACHINE Metabase
DEFINITIONS
  PropertiesInfo_struct == struct(
        DBID: INT,
        PropID: INT,
        Name: seq(0..255),
        Description: seq(0..255),
        WWWTemplatePage: seq(0..255),
        UpdateStatus: INT
  )
ABSTRACT_VARIABLES
  PropertiesInfo
INVARIANT
  PropertiesInfo: FIN(PropertiesInfo_struct) &
  !(prop1, prop2).(
     prop1: PropertiesInfo & prop2: PropertiesInfo &
     prop1'DBID = prop2'DBID & prop1'PropID = prop2'PropID =>
     prop1 = prop2) &
INITIALISATION
  PropertiesInfo:= {}
OPERATIONS
result <-- getNewSystemID(adbid) =
PRE adbid: INT
THEN
    IF card({ system | system: SystemInfo &
                       system'DBID = adbid }) = 0
    THEN
        result:= 1
    ELSE
        result:= max({ systemid | systemid: INT &
            #(system).(system: SystemInfo &
                       system'DBID = adbid &
                       system'SystemID = systemid) }) + 1
    END
END
```

Можно видеть, что семантика целевой реляционной схемы определяется аналогично семантике реляционной схемы источника (раздел 1.1). Отметим, что в таблицах целевой

базы данных присутствует ключевой атрибут DBID. Генерация значений этого атрибута производится веб-сервисом, вызывающим процедуры целевой базы данных. Семантика генерации значений выражается операцией `getNewSystemID` спецификации `Metabase`.

Для обновления данных в интегрированной базе данных определены процедуры. Например, процедура обновления свойств веществ выглядит следующим образом:

```
CREATE PROCEDURE [dbo].[UpdatePropertiesInfo]
@DBID int, @PropID int, @Name varchar(256), @Description text,
@WWWTemplatePage varchar(256), @UpdateStatus int
AS
IF @UpdateStatus=2
BEGIN
  DELETE FROM PropertiesInfo WHERE DBID=@DBID AND PropID=@PropID
END
ELSE
BEGIN
  IF EXISTS (SELECT DBID FROM PropertiesInfo
             WHERE DBID=@DBID AND PropID=@PropID)
    UPDATE PropertiesInfo
    SET [Name]=@Name, [Description]=@Description,
        WWWTemplatePage=@WWWTemplatePage, UpdateStatus=1
    WHERE DBID=@DBID AND PropID=@PropID
  ELSE
    INSERT INTO PropertiesInfo (DBID, PropID, [Name],
        [Description], WWWTemplatePage, UpdateStatus)
    VALUES (@DBID, @PropID, @Name, @Description,
            @WWWTemplatePage, 1)
  END
RETURN 0
```

Статус обновления записи (`UpdateStatus`), равный 2, означает, что запись должна быть удалена. Если запись уже существует в интегрированной базе данных (это определяется по значениям ключевых атрибутов `DBID`, `PropID`), то обновляются значения ее атрибутов. В противном случае запись добавляется в базу данных. Семантика этой процедуры выражается операцией машины `Metabase`:

```
UpdatePropertiesInfo(adbid, apropid, aname, adescription,
                     awwwtemplatepage, anupdatestatus) =
PRE
    adbid: INT &
    apropid: INT &
    aname: seq(0..255) &
    adescription: seq(0..255) &
    awwwtemplatepage: seq(0..255) &
    anupdatestatus: INT
THEN
    IF anupdatestatus = 2
    THEN
```

```
        PropertiesInfo := PropertiesInfo -
         { rcrd | rcrd: PropertiesInfo  & rcrd'DBID = adbid &
                  rcrd'PropID  = apropid}
    ELSE
        IF #(rcrd).(rcrd: PropertiesInfo & rcrd'DBID = adbid &
                    rcrd'PropID = apropid)
        THEN // Update
             PropertiesInfo :=
             (PropertiesInfo - { rcrd | rcrd: PropertiesInfo &
                rcrd'DBID = adbid & rcrd'PropID = apropid}) \/
             {rec(DBID: adbid, PropID: apropid, Name: aname,
                  Description: adescription,
                  WWWTemplatePage: awwwtemplatepage,
                  UpdateStatus: 1)}
        ELSE // Insert
             PropertiesInfo := PropertiesInfo \/
             {rec(DBID: adbid, PropID: apropid, Name: aname,
                  Description: adescription,
                  WWWTemplatePage: awwwtemplatepage,
                  UpdateStatus: 1)}
        END
    END
END;
```

Здесь семантика операции DELETE выражается операцией разности множеств «—», семантика SELECT – операцией выделения множества «{ | }», семантика операции UPDATE – композицией операций разности и объединения множеств («\/»), семантика INSERT – операцией объединения множеств, семантика EXISTS – формулой с квантором существования «#».

1.4. Семантика императивной программы интеграции данных. Фрагмент программы интеграции данных из базы данных Bandgap в интегрированной базе данных ИМЕТ приведен в файле *UpdateBandGapMeta-ver.vbs.* В настоящем разделе мы ограничимся обсуждением операции манипулирования данными о свойствах вещества (как в разделах 1.1, 1.2 и 1.3).

Для загрузки обновлений данных о свойствах веществ из источника и их преобразования в промежуточное представление служит функция `ProcessPropertiesInfo` (приведен фрагмент функции):

```
function ProcessPropertiesInfo(objRootElement)
Dim tmp, tmpLink, tmpLink2, theDate
Dim NOMPROP, UpdateStatus, NAZVPROP, HTML
  RSN.Open "SELECT * FROM _PropertiesConv WHERE UpdateStatus>0"
  if NOT RSN.EOF Then
    Set tmp = objXML.createNode(1, "PropertiesInfo", "")
    objRootElement.appendChild(tmp)
  end if
```

```
  Do while NOT RSN.EOF
    NOMPROP = RSN("NOMPROP")
    ...
    Set tmpLink = objXML.createNode(1, "PropertiesInfoItem", "")
    Set tmpLink2 = objXML.createNode(1, "PropID", "")
    tmpLink2.text = NOMPROP
    tmpLink.appendChild(tmpLink2)
    ...
    tmp.appendChild(tmpLink)
    RSN.MoveNext
  Loop
end function
```

Эта функция извлекает обновленные данные из таблицы `_PropertiesConv`, для каждой записи извлеченных данных в цикле создает элемент `PropertiesInfoItem` и присваивает его свойствам (например, `PropID`) соответствующие значения из записи (например, `NOMPROP`).

Семантика программы интеграции данных выражается AMN-спецификацией `Bandgap2MetabaseRef`, включающей (INCLUDES [8]) семантические спецификации схем источника, промежуточного представления и интегрированной базы данных:

```
REFINEMENT Bandgap2MetabaseRef
INCLUDES Bandgap, MUService, Metabase
```

Управление последовательностью исполнения операций трансформации данных осуществляется при помощи переменной `state`, принимающей значения из множества `TRANSFORMATION_PERFORMED`:

```
SETS
    TRANSFORMATION_PERFORMED = {
        READY_TO_TRANSFORM,
        TRANSFORM_PROPERTIESCONV,
        TRANSFORM_PROPERTIESCONV_RECORD,
        TRANSFORM_PROPERTIES_INFO,
        TRANSFORM_PROPERTIES_INFO_NODE,
        ...
    }
ABSTRACT_VARIABLES state
INVARIANT
    state: TRANSFORMATION_PERFORMED
INITIALISATION
    state:= READY_TO_TRANSFORM
```

Так, например, состояние TRANSFORM_PROPERTIESCONV означает, что происходит трансформация записей таблицы `PropertiesConv` в промежуточное представление, а состояние TRANSFORM_PROPERTIES_INFO – что происходит загрузка данных из XML-элементов `PropertiesInfoItem` в целевую базу данных.

Определяются вспомогательные переменные, соответствующие множествам еще не обработанных записей или элементов:

```
ABSTRACT_VARIABLES
    RSN_PropertiesConv,
    propertiesInfo
INVARIANT
    RSN_PropertiesConv: FIN(PropertiesConv_struct) &
    propertiesInfo: FIN(PropertiesInfoItem)
INITIALISATION
    RSN_PropertiesConv:= {} ||
    propertiesInfo:= {}
```

Семантика операций трансформации отдельных записей или XML-элементов выражается операциями AMN (ниже приведен пример одной из операций, соответствующей преобразованию записи из таблицы `PropertiesConv` вышеприведенной функцией `ProcessPropertiesInfo`):

```
ProcessPropertiesInfoItem =
SELECT state = TRANSFORM_PROPERTIESCONV_RECORD
THEN
  VAR NOMPROP, UpdateStatus, NAZVPROP, HTML, tmpLink IN
    IF RSN_PropertiesConv /= {} THEN
      ANY rcrd WHERE rcrd: RSN_PropertiesConv THEN
        NOMPROP := rcrd'NOMPROP;
        UpdateStatus := rcrd'UpdateStatus;
        NAZVPROP := rcrd'NAZVPROP;
        HTML := rcrd'HTML;
        tmpLink := rec(PropID: NOMPROP, Name: NAZVPROP,
          Description: [0], WWWTemplatePage: HTML,
          UpdateStatus: UpdateStatus);
        createNodePropertiesInfoItem(tmpLink);
        RSN_PropertiesConv:= RSN_PropertiesConv - {rcrd}
      END
    ELSE
      state:= TRANSFORM_DBCONTENT
    END
  END
END;
```

2. Верификация программ интеграции данных

Свойства корректности программ интеграции данных, подлежащих верификации, определяются экспертом вручную в виде формул, конъюнктивно присоединяемых в раздел INVARIANT. Например, свойство корректности интеграции данных, относящихся к свойствам веществ, выглядит следующим образом:

```
state = READY_TO_TRANSFORM &
MetaBase /= MetaBaseConst &
(MetaBase'SystemInfo /= {} or MetaBase'PropertiesInfo /= {} or
 MetaBase'DBContent /= {}) =>
```

```
!(record).(record: PropertiesConv & record'UpdateStatus = 2 =>
    not(#(prop).(prop: PropertiesInfo &
        record'NOMPROP = prop'PropID)) ) &
!(record).(record: PropertiesConv & record'UpdateStatus > 0 &
    record'UpdateStatus /= 2 =>
    #(prop).(prop: PropertiesInfo &
            record'NOMPROP = prop'PropID &
            record'NAZVPROP = prop'Name &
            record'HTML = prop'WWWTemplatePage))
```

Формула утверждает, что после выполнения всех операций трансформации данных в целевой базе данных нет записей о свойствах веществ, помеченных на удаление в исходной базе данных, и есть записи о всех свойствах веществ, помеченных на обновление или вставку в исходной базе данных.

Полная формула, выражающая корректность интеграции исходных данных базы Bandgap в целевую интегрированную базу данных ИМЕТ, была добавлена в раздел INVARIANT спецификации `Bandgap2MetabaseRef`.

Спецификации *Bandgap.mch, MUService.mch, Metabase.mch, Bandgap2Metabase.ref* были загружены в проект инструментария Atelier B, автоматически были сгенерированы теоремы корректности спецификаций, применены средства автоматического и интерактивного доказательств с участием эксперта. Статистика доказательств приведена в табл. 1.

Табл. 1. Количество сгенерированных и автоматически доказанных теорем
Table 1. Number of generated and automatically proven theorems

Спецификация	Количество сгенерированных теорем	Количество автоматически доказанных теорем при режиме доказательства Automatic			
		Force Fast	Force 0	Force 1	Force 2
Bandgap.mch	6	3	6		
MUService.mch	8				2
Metabase.mch	33	5	22	23	
Bandgap2Metabase.ref	320	8	267	272	

Заключение

Рассмотрен подход к верификации корректности интеграции данных в интегрированной системе баз данных по свойствам неорганических веществ и материалов Института металлургии и материаловедения им. А.А. Байкова РАН. Интеграция данных из исходных реляционных баз данных проводится в системе в два этапа с использованием промежуточного XML-представления. Программы интеграции данных выражаются с использованием композиции императивного языка программирования Visual Basic и декларативного языка реляционных баз данных SQL. Программы интеграции предназначены для регулярного обновления данных из источников в интегрированной системе.

Разработанный подход основан на выражении семантики схем источников данных, промежуточного представления, целевой базы данных, а также программ интеграции данных в формальном языке спецификаций AMN, основанном на логике предикатов первого порядка и теории множеств. Подход проиллюстрирован на примере интеграции в системе базы данных о ширине *запрещенной зоны* (характеризует природу химической связи в материале) основных классов неорганических веществ. Автоматизированные средства доказательства применены для автоматической генерации и доказательства теорем корректности интеграции данных.

Отметим, что на текущем этапе исследований AMN-спецификации, выражающие семантику схем данных и программ интеграции данных, создаются экспертом вручную. Для повышения автоматизации и применимости подхода генерацию AMN-спецификаций следует автоматизировать (это является наиболее важным направлением будущей работы). На основании опыта автора по разработке автоматизированных средств определения формальной семантики моделей данных это можно сделать с использованием технологий движимой моделями инженерии (MDE). При этом синтаксис моделей данных и языков описания программ интеграции данных представляется в метамодели *Ecore* [16], а генерация семантических AMN-спецификаций реализуется на языке трансформации моделей ATL [17]. Вычислительная сложность генерации AMN-спецификаций при этом обычно линейна от размера исходных схем и программ интеграции даже в тех случаях, когда схемы содержат множество взаимосвязанных типов сущностей. Однако при этом основные временные затраты на верификацию приходятся не на работу автоматических средств генерации или доказательства, а на ручную работу эксперта по формулировке свойств программ интеграции, подлежащих проверке, и интерактивному доказательству теорем корректности, оставшихся недоказанными после применения автоматических средств доказательства. Такие издержки характерны при применении формальной верификации сложных систем, но они значительно меньше издержек на исправление ошибок в программах, выпущенных в производство [5].

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Conflicts of Interest. The authors declare no conflicts of interest.

Литература

1. *Masmoudi M., Ben Abdallah Ben Lamine S., Karray M.H., Archimede B., Zghal H.B.* Semantic data integration and querying: A survey and challenges // ACM Comput. Surv. 2024. V. 56, No 8. Art. 209. https://doi.org/10.1145/3653317.

2. The Binary Star Database. ИНИСАН, 2025. https://bdb.inasan.ru/.

3. Крупный научный проект фундаментальных исследований «Актуальные научные задачи стратегии адаптации потенциала землепользования России в современных условиях беспрецедентных вызовов (экономический кризис, изменения климата, кризис глобальных тенденций природопользования)». Почвенный институт им. В.В. Докучаева, 2023. https://www.esoil.ru/activities/projects_programs/minobr/knp_2020/.

4. Интегрированная система баз данных по свойствам неорганических веществ и материалов. ИМЕТ РАН, 2025. https://imet-db.ru/.

5. *White N., Matthews S., Chapman R.* Formal verification: Will the seedling ever flower? // Phil. Trans. R. Soc. A. 2017. V. 375, No 2104. Art. 20150402. https://doi.org/10.1098/rsta.2015.0402.

6. *Guagliardo P., Libkin L.* A formal semantics of SQL queries, its validation, and applications // Proc. VLDB Endowment (PVLDB). 2017. V. 11, No 1. P. 27–39. https://doi.org/10.14778/3151113.3151116.

7. *Rahim L.Ab., Whittle J.* A survey of approaches for verifying model transformations // Software Syst. Model. 2015. V. 14, No 2. P. 1003–1028. https://doi.org/10.1007/s10270-013-0358-0.

8. *Abrial J.-R.* The B-Book: Assigning Programs to Meanings. New York, NY: Cambridge Univ. Press, 1996. xxxiv, 779 p. https://doi.org/10.1017/CBO9780511624162.

9. Atelier B: The industrial tool to efficiently deploy the B Method. 2025. http://www.atelierb.eu/.

10. *Butler M., Körner P., Krings S., Lecomte T., Leuschel M., Mejia L.-F., Voisin L.* The first twenty-five years of industrial use of the B-Method // ter Beek M.H., Ničković D. (Eds.) Formal Methods for Industrial Critical Systems (FMICS 2020). Ser.: Lecture Notes in Computer Science. V. 12327. Cham: Springer, 2020. P. 189–209. https://doi.org/10.1007/978-3-030-58298-2_8.

11. *Stupnikov S.* Semantics and verification of entity resolution and data fusion operations via transformation into a formal notation // Kalinichenko L., Kuznetsov S., Manolopoulos Y. (Eds.) Data Analytics and Management in Data Intensive Domains (DAMDID/RCDL 2016). Ser.: Communications in Computer and Information Science. V. 706. Cham: Springer, 2017. P. 145–162. https://doi.org/10.1007/978-3-319-57135-5_11.

12. *Stupnikov S.A.* Query-driven verification of data integration in the RDF data model // Lobachevskii. J. Math. 2023. V. 44, No 1. P. 205–218. https://doi.org/10.1134/S1995080223010389.

13. *Kiselyova N.N., Dudarev V.A., Zemskov V.S.* Computer information resources of inorganic chemistry and materials science // Russ. Chem. Rev. 2010. V. 79, No 2. P. 145–166. https://doi.org/10.1070/RC2010v079n02ABEH004104.

14. *Kiselyova N.N., Dudarev V.A., Stolyarenko A.V.* Integrated system of databases on the properties of inorganic substances and materials // High Temp. 2016. V. 54, No 2. P. 215–222. https://doi.org/10.1134/S0018151X16020085.

15. *Kiselyova N.N., Dudarev V.A., Korzhuyev M.A.* Database on the bandgap of inorganic substances and materials // Inorg. Mater.: Appl. Res. 2016. V. 7, No 1. P. 34–39. https://doi.org/10.1134/S2075113316010093.

16. *Steinberg D., Budinsky F., Paternostro M., Merks E.* EMF: Eclipse Modeling Framework. The Eclipse Ser. Addison-Wesley, 2008. 744 p.

17. ATL – a model transformation technology. 2025. https://eclipse.org/atl/.

References

1. Masmoudi M., Ben Abdallah Ben Lamine S., Karray M.H., Archimede B., Zghal H.B. Semantic data integration and querying: A survey and challenges. *ACM Comput. Surv.*, 2024, vol. 56, no. 8, art. 209. https://doi.org/10.1145/3653317.

2. The Binary Star Database. INASAN, 2025. https://bdb.inasan.ru/.

3. Major Basic Research Project. Current scientific tasks in adapting Russia's land use potential to unprecedented global challenges (economic crisis, climate change, and crisis in global natural resource management). V.V. Dokuchaev Soil Sci. Inst., 2023. https://www.esoil.ru/activities/projects_programs/minobr/knp_2020/. (In Russian)

4. Integrated system of databases on the properties of inorganic substances and materials. A.A. Baikov Inst. Metall. Mater. Sci., Russ. Acad. Sci., 2005. https://imet-db.ru/. (In Russian)

5. White N., Matthews S., Chapman R. Formal verification: Will the seedling ever flower? *Philos. Trans. R. Soc. A*, 2017, vol. 375, no. 2104, art. 20150402. https://doi.org/10.1098/rsta.2015.0402.

6. Guagliardo P., Libkin L. A formal semantics of SQL queries, its validation, and applications. *Proc. VLDB Endowment* (*PVLDB*), 2017, vol. 11, no. 1, pp. 27–39. https://doi.org/10.14778/3151113.3151116.

7. Rahim L.Ab., Whittle J. A survey of approaches for verifying model transformations. *Software Syst. Model.*, 2015, vol. 14, no. 2, pp. 1003–1028. https://doi.org/10.1007/s10270-013-0358-0.

8. Abrial J.-R. *The B-Book: Assigning Programs to Meanings*. New York, NY, Cambridge Univ. Press, 1996. xxxiv, 779 p. https://doi.org/10.1017/CBO9780511624162.

9. Atelier B: The industrial tool to efficiently deploy the B Method. 2025. http://www.atelierb.eu/.

10. Butler M., Körner P., Krings S., Lecomte T., Leuschel M., Mejia L.-F., Voisin L. The first twenty-five years of industrial use of the B-Method. In: ter Beek M.H., Ničković D. (Eds.) *Formal Methods for Industrial Critical Systems* (*FMICS 2020*). Ser.: Lecture Notes in Computer Science. Vol. 12327. Cham, Springer, 2020, pp. 189–209. https://doi.org/10.1007/978-3-030-58298-2_8.

11. Stupnikov S. Semantics and verification of entity resolution and data fusion operations via transformation into a formal notation. In: Kalinichenko L., Kuznetsov S., Manolopoulos Y. (Eds.) *Data Analytics and Management in Data Intensive Domains* (*DAMDID/RCDL 2016*). Ser.: Communications in Computer and Information Science. Vol. 706. Cham, Springer, 2017, pp. 145–162. https://doi.org/10.1007/978-3-319-57135-5_11.

12. Stupnikov S.A. Query-driven verification of data integration in the RDF data model. *Lobachevskii. J. Math.*, 2023, vol. 44, no. 1, pp. 205–218. https://doi.org/10.1134/S1995080223010389.

13. Kiselyova N.N., Dudarev V.A., Zemskov V.S. Computer information resources of inorganic chemistry and materials science. *Russ. Chem. Rev.*, 2010, vol. 79, no. 2, pp. 145–166. https://doi.org/10.1070/RC2010v079n02ABEH004104.

14. Kiselyova N.N., Dudarev V.A., Stolyarenko A.V. Integrated system of databases on the properties of inorganic substances and materials. *High Temp.*, 2016, vol. 54, no. 2, pp. 215–222. https://doi.org/10.1134/S0018151X16020085.

15. Kiselyova N.N., Dudarev V.A., Korzhuyev M.A. Database on the bandgap of inorganic substances and materials. *Inorg. Mater.: Appl. Res.*, 2016, vol. 7, no. 1, pp. 34–39. https://doi.org/10.1134/S2075113316010093.

16. Steinberg D., Budinsky F., Paternostro M., Merks E. *EMF: Eclipse Modeling Framework*. The Eclipse Ser. Addison-Wesley, 2008. 744 p.

17. ATL – a model transformation technology. 2025. https://eclipse.org/atl/.

Информация об авторах

Сергей Александрович Ступников, ведущий научный сотрудник, Федеральный исследовательский центр «Информатика и управление» Российской академии наук

E-mail: *sstupniukov@ipiran.ru*

ORCID: https://orcid.org/0000-0003-4720-8215

Author Information

Sergey A. Stupnikov, Leading Researcher, Federal Research Center "Computer Science and Control", Russian Academy of Sciences

E-mail: *sstupniukov@ipiran.ru*

ORCID: https://orcid.org/0000-0003-4720-8215

Поступила в редакцию 1.03.2025 Received March 1, 2025

Принята к публикации 23.03.2025 Accepted March 23, 2025