

Figure 1 — Data export process

NodePolicy is used to intercept events for creating, updating and deleting files (nodes) in Alfresco. After calling the method, the document's metadata is saved as a transaction resource, which allows it to be read when the transaction commits. TransactionListener is also an extension point of Alfresco. In this class, the event node is saved in alfresco, prepares data for dispatch using the Serializer class and invokes parallel dispatch of messages to the queue. SyncEventStorage encapsulates the logic for interacting with event nodes and provides an interface for creating, reading and deleting event nodes. Serializer is used for filtering metadata and converting to a certain format. ParallelPublisher implements the Facade design pattern and encapsulates a thread pool and is used to send messages asynchronously using the Publisher class. A separate Alfresco PollerJob extension point implements the logic for resending messages to the queue. It starts at a certain interval and reads all event nodes and calls ParallelPublisher. After the message has been sent, it is deleted. Thus, only unsent event nodes are in the SyncEventStorage.

**Mikhail Bouzdalkin**

(Fr. Skorina GSU, Gomel)

Scientific adviser **Viktar Liauchuk**, Ph.D. in technics, associate professor

## INTEGRATION TESTING OF THE DATA EXPORT SUBSYSTEM

Integration testing allows to check the correctness of the interaction of software interfaces as a whole, which is impossible in the case of unit testing.

Since in given case there is a specific enterprise project, the export subsystem includes a replicator that receives data from the queue and sends data to a specific repository. The JUnit framework, Testcontainers Java library and Spring JmsTemplate were chosen for the integration testing.

Spring JmsTemplate allows to emulate messages coming from the queue, which are subsequently processed by the business logic of the replication service, and then the data is sent to the Allegrograph repository located in the Docker container (Figure 1). This container starts working before the first test starts and is deleted after all tests are completed. Testcontainers library allows to manage lifecycle of Allegrograph container, set up version, image name, registry, ports etc. After the data has been saved in the Allegrograph, it must be subtracted using a SPARQL query and compared with the original data. In order to reduce reliance on serialization differences, Jena model comparison was used instead of string values.

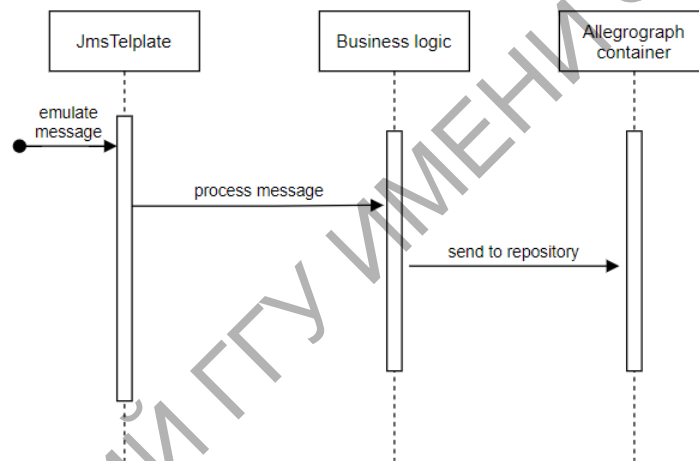


Figure 1 – Interaction scheme

**Stepan Dubrovski**

(Fr. Skorina GSU, Gomel)

Scientific advisor **Viktar Liauchuk**, Ph.D. in technics, associate professor

## FUNCTIONAL OPPORTUNITIES OF A MOBILE CLIENT FOR A DISTRIBUTION NETWORK

Nowadays, the way of gaining access to a certain product or service has changed a lot, even over several years. Today it is impossible to conduct a business focused on the distribution of goods or the provision of services without using modern approaches to sales and consumption. Currently, retail