Asynchronous Resources - ActiveMQ

Pavol Mederly
May 2021

Agenda

- What
- When
- How
- A practical example
- Discussion

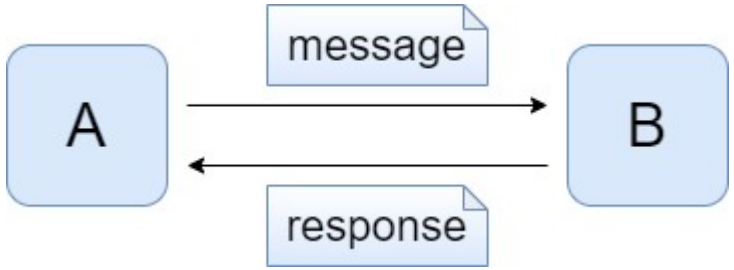
Evolveum

What are Asynchronous Resources?

Synchronous vs. Asynchronous Communication

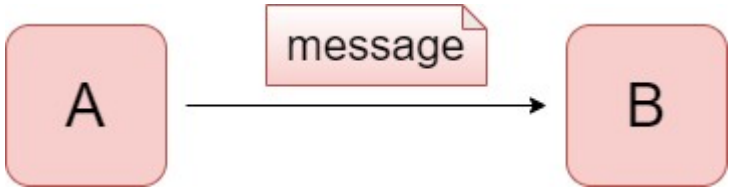
Synchronous

- A party sends a message and **waits for reply**.
- **Only then** it continues with its business.



Asynchronous

- A party sends a message and **does not wait for reply**. (Or does not expect reply at all.)
- It continues with its business **immediately**.



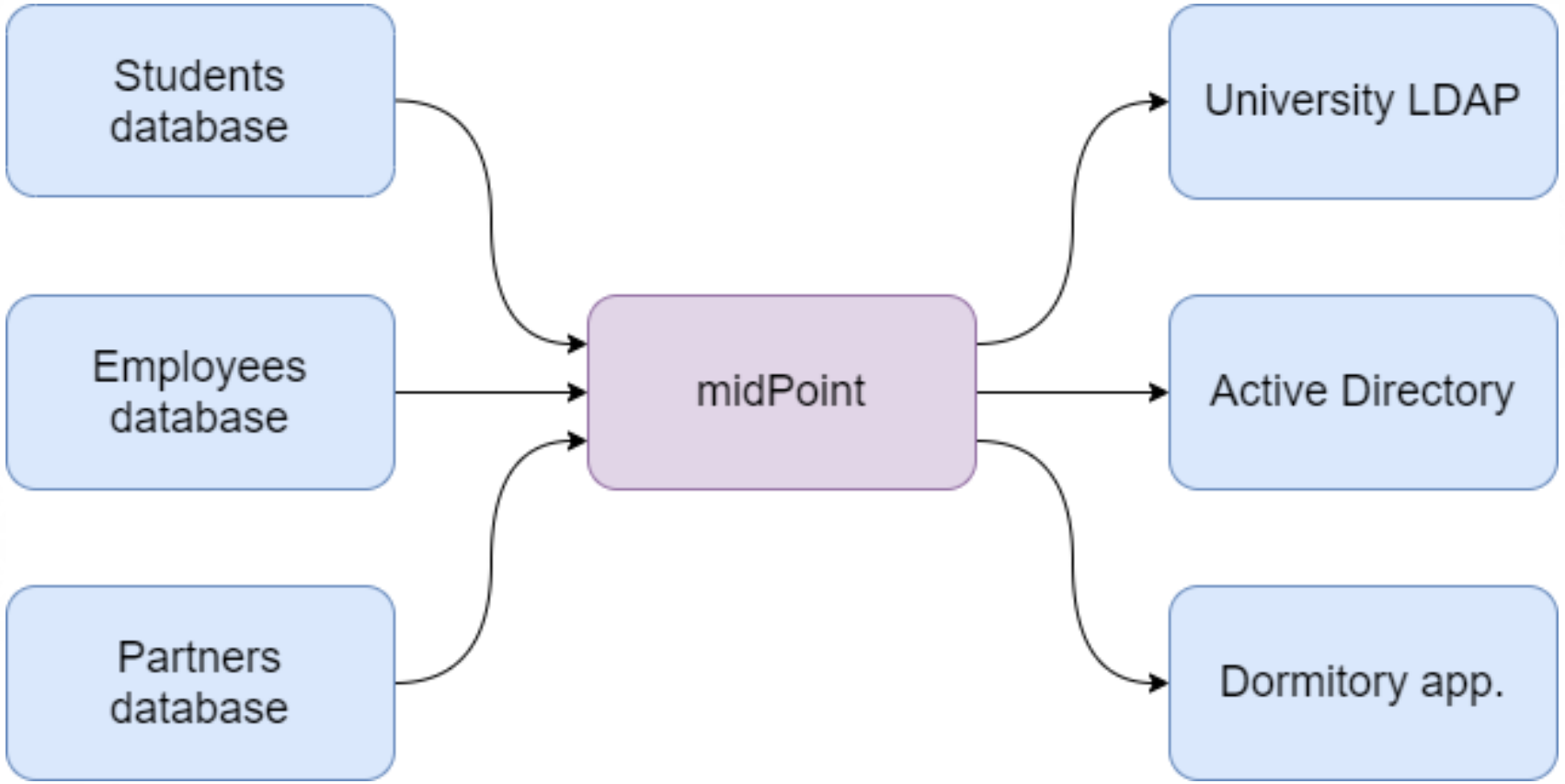
Evolveum

Synchronous vs. Asynchronous Communication in MidPoint

	Synchronous Communication	Asynchronous Communication
From Sources	Reconciliation Live Synchronization	Asynchronous Update
To Targets	Standard Provisioning (Reconciliation)	Asynchronous Provisioning

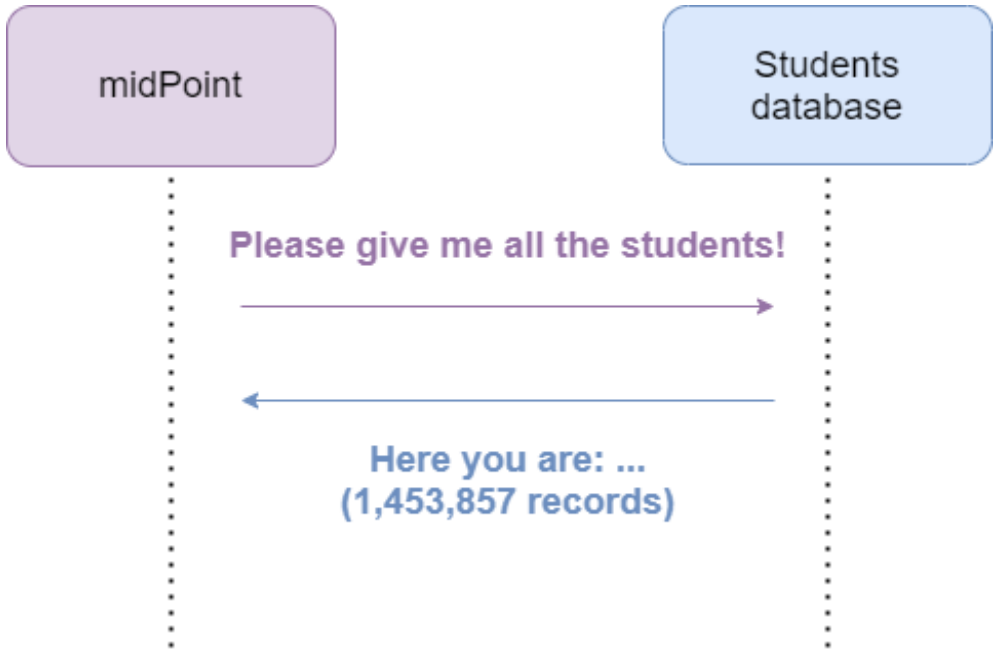
Evolveum

Sample Scenario



Evolveum

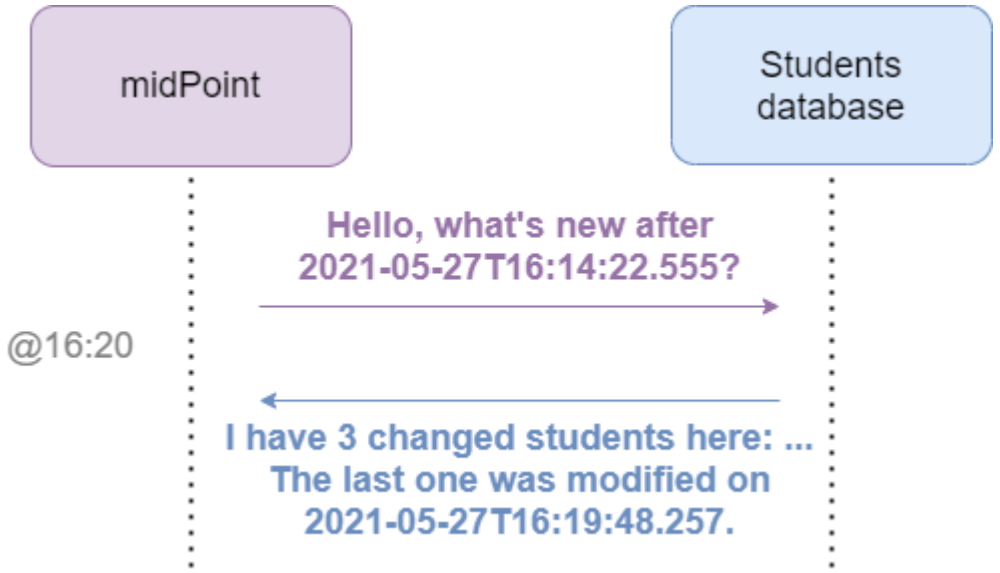
Source Side: Reconciliation



Note: slightly simplified

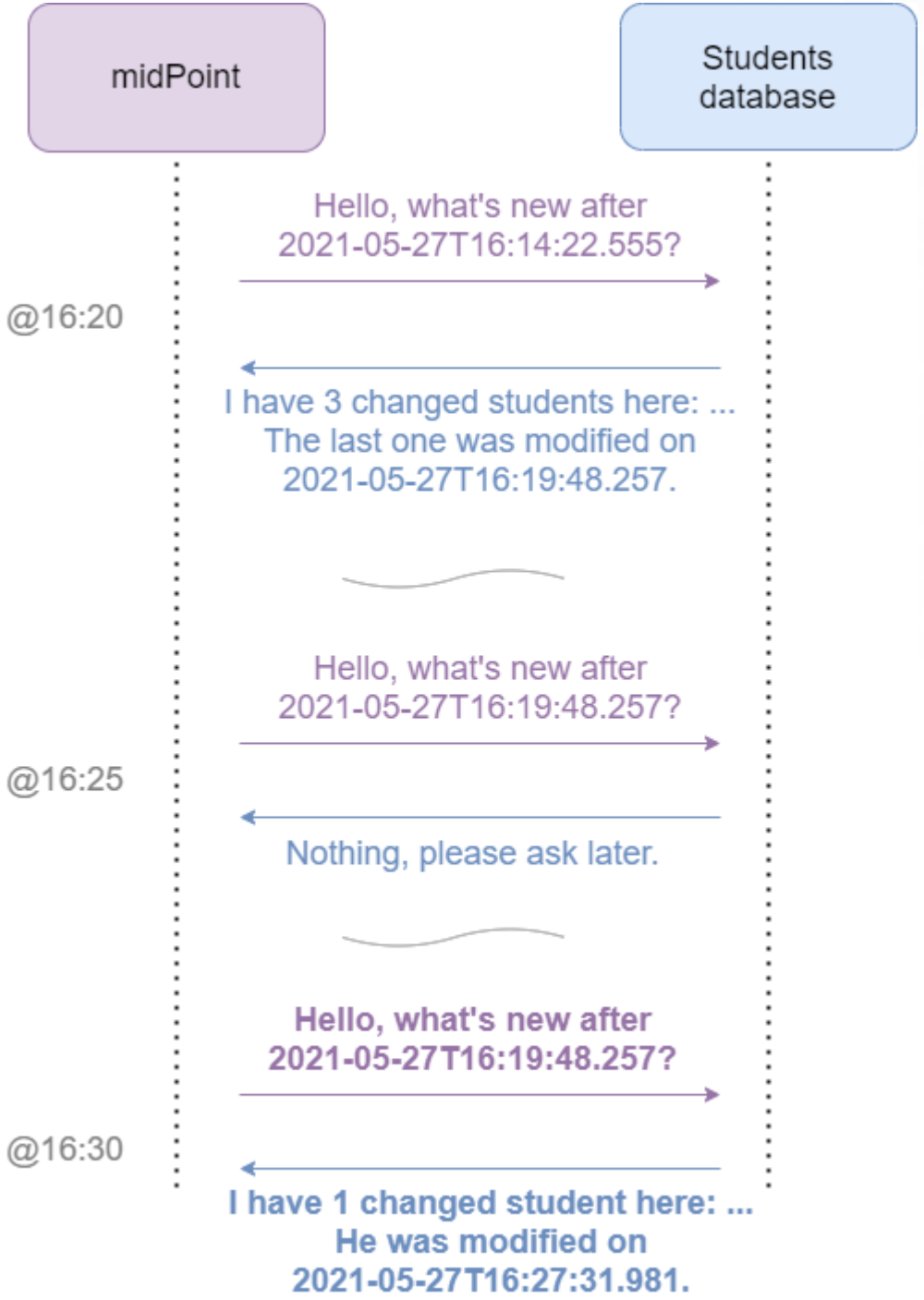
Evolveum

Source Side: Live Sync



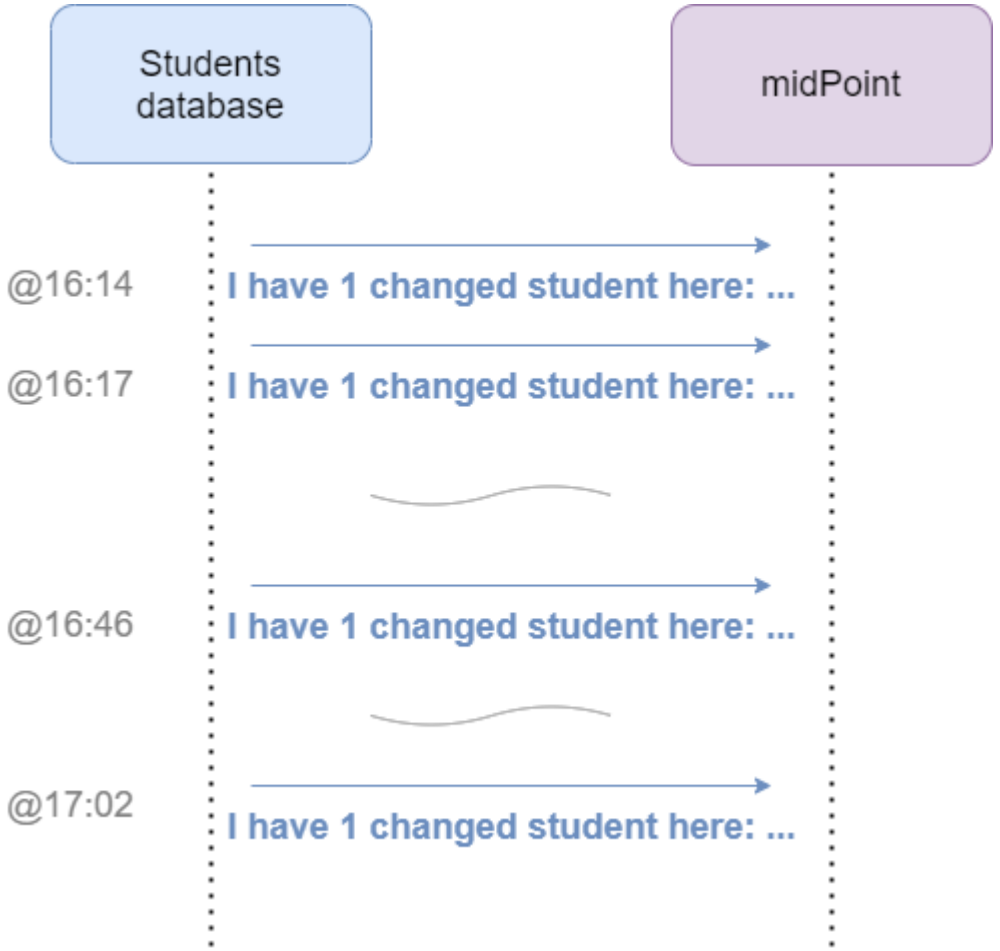
Evolveum

Live Sync



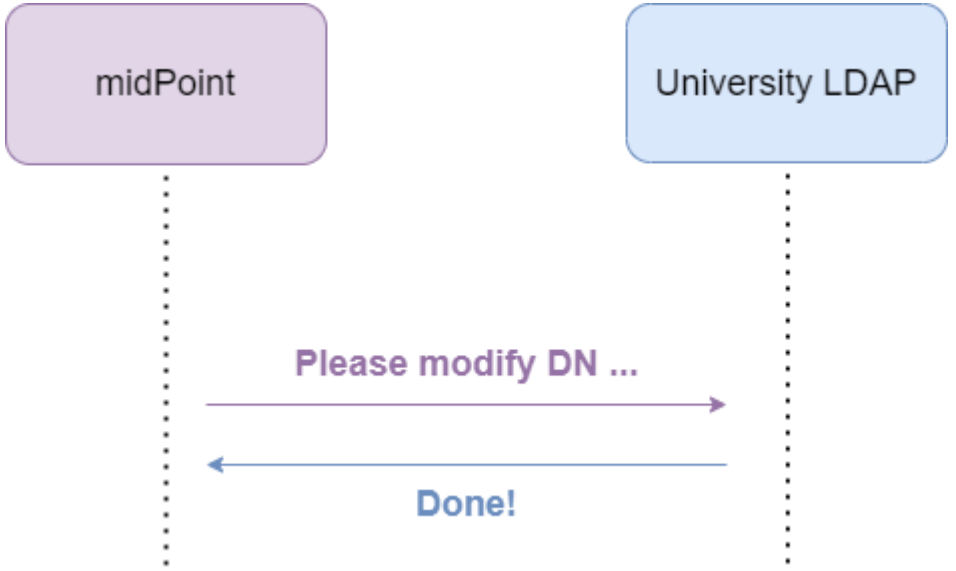
Evolveum

Source Side: Asynchronous Updates



Evolveum

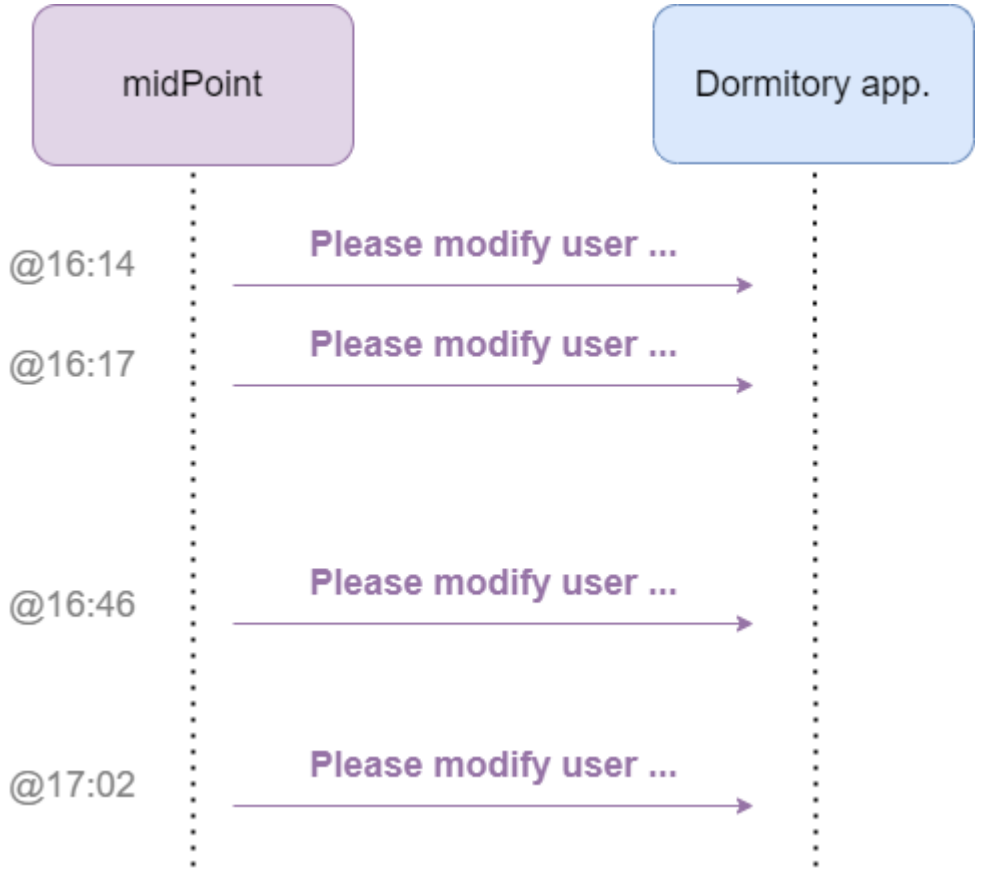
Target Side: Standard Provisioning



The synchronous nature of communication enables immediate reaction to unexpected situations.

Evolveum

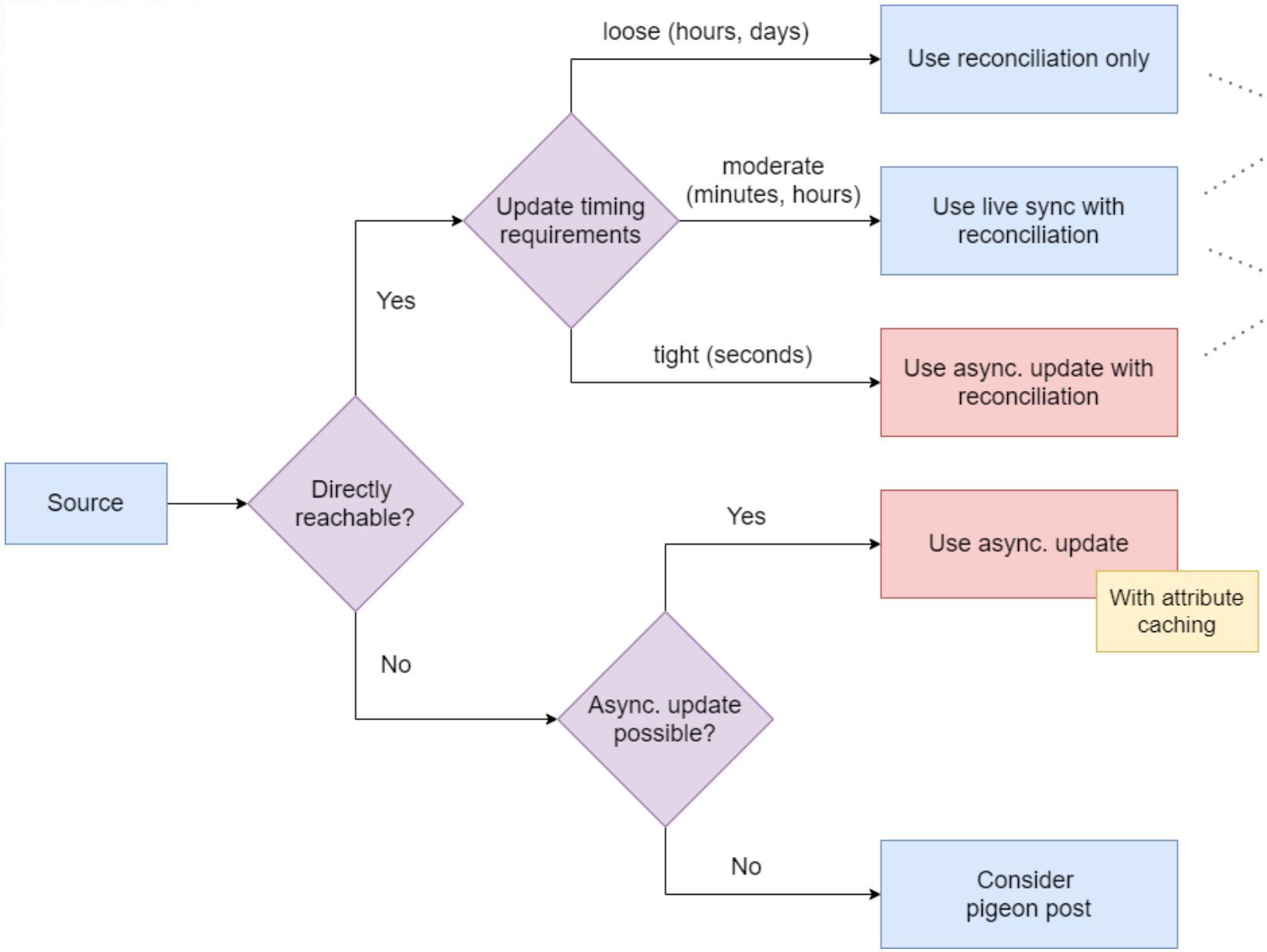
Target Side: Asynchronous Provisioning



Evolveum

When To Use Asynchronous Resources?

Deciding at the Source Side

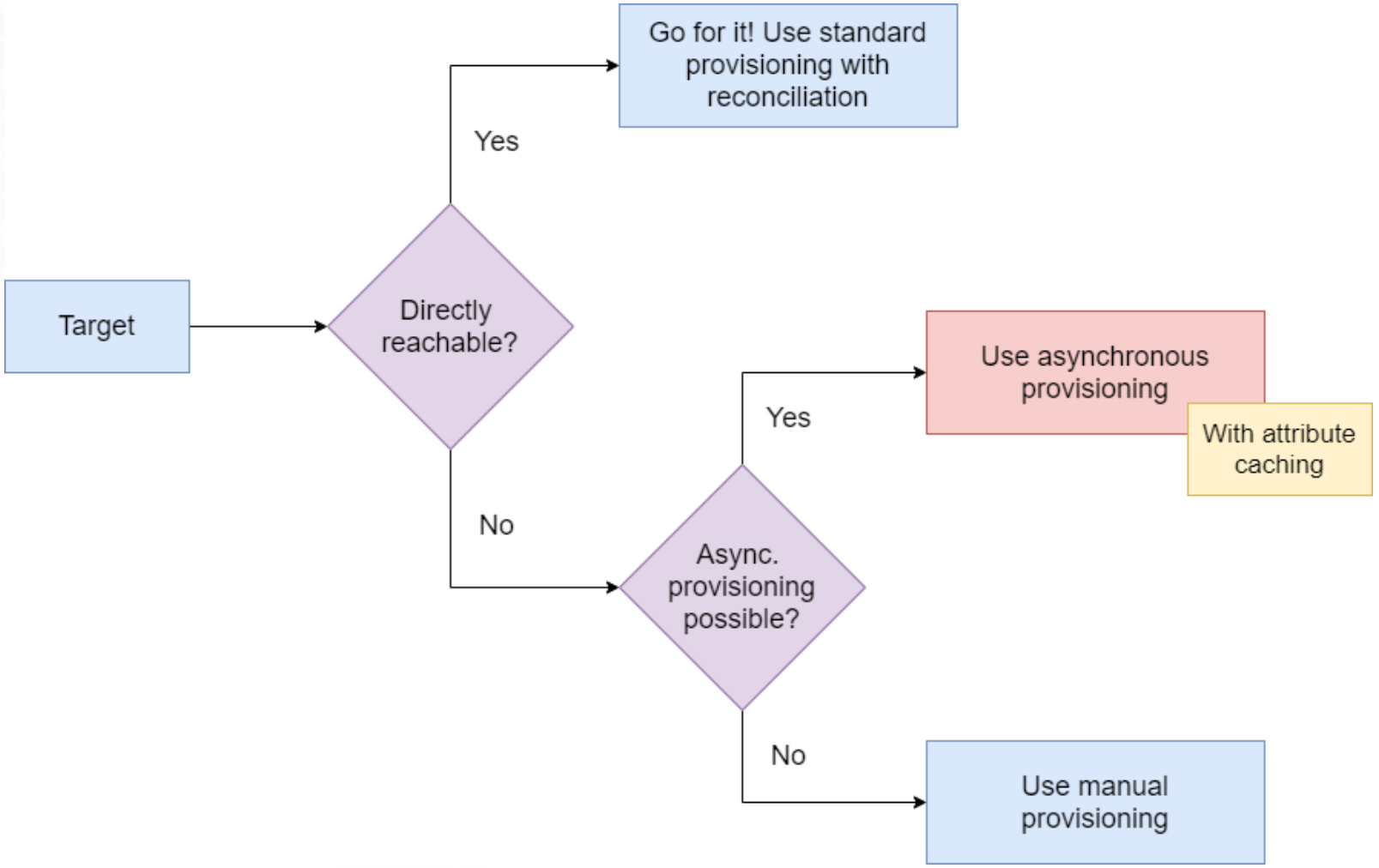


Costs:

- *Experimental nature*
- *Higher complexity*
- *Lack of reconciliation**
- *Need of attribute caching**

** when not directly reachable*

Deciding at the Target Side



Note: A special case could be a resource with very slow responses. Here the asynchronous provisioning could make sense as well (in the future).

Costs:

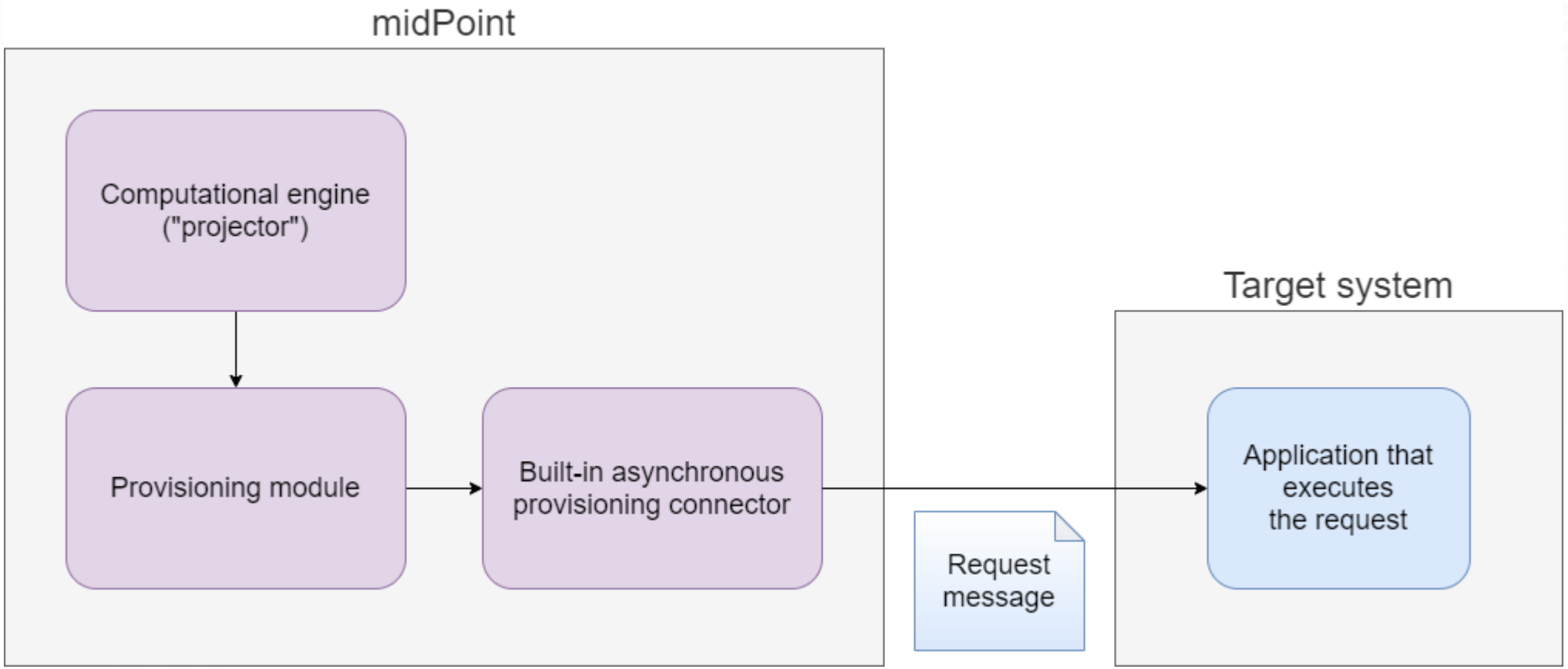
- *Cannot immediately react to exceptional situations*
- *Experimental nature*
- *Higher complexity*
- *Need of attribute caching**
- *Lack of reconciliation**

** currently*

How To Use Asynchronous Provisioning?

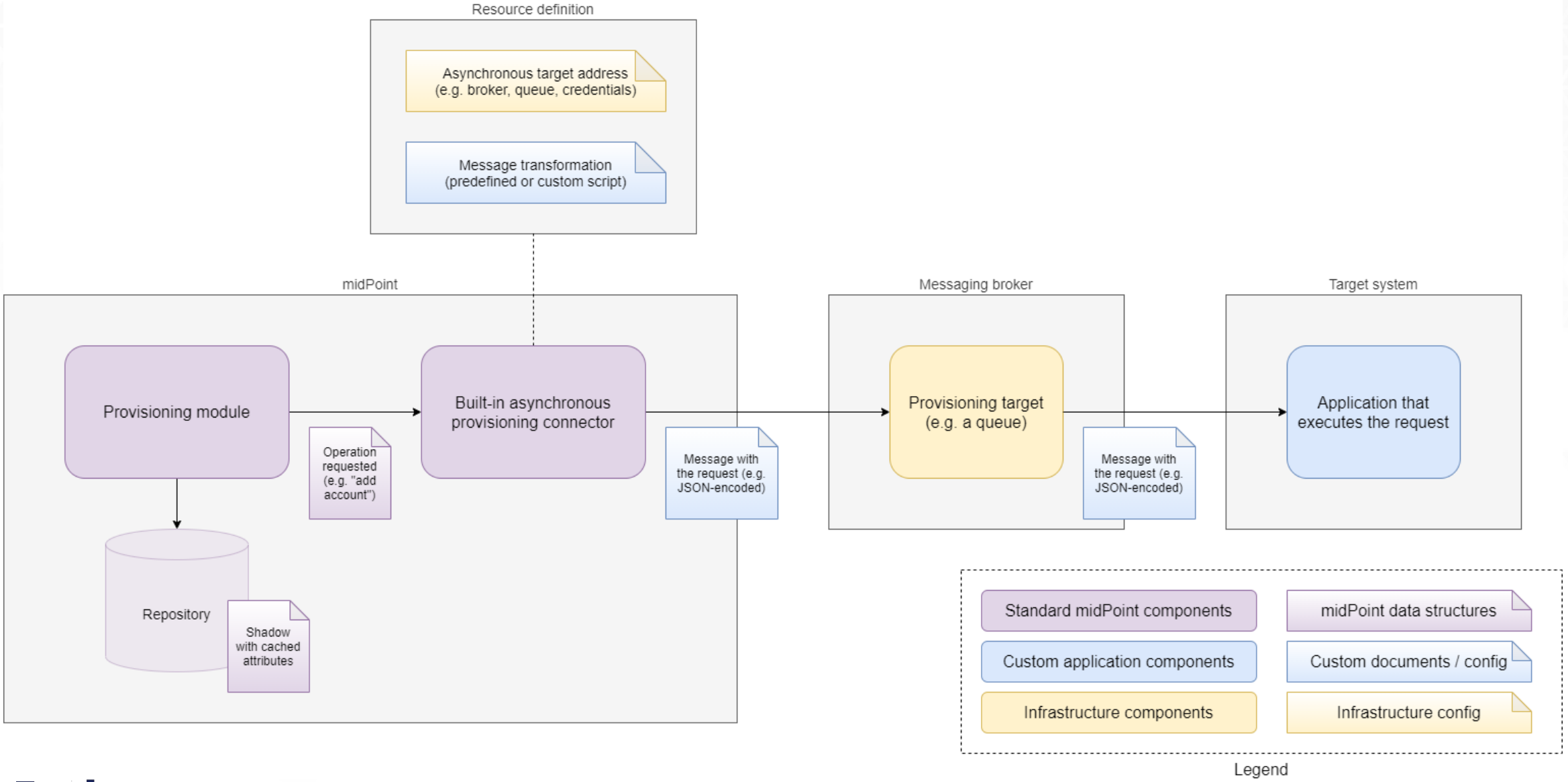
Note: Details about asynchronous update (source side) are not in the scope of this webinar.

Overall Schema Simplified



Evolveum

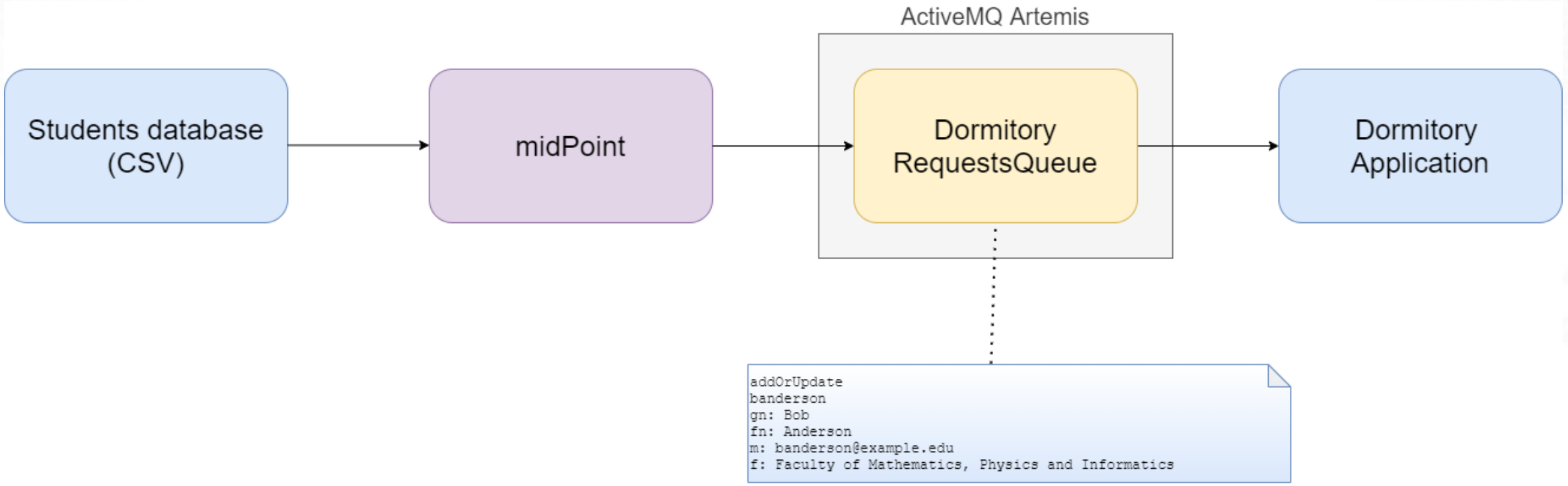
Overall Schema Details



Evolveum

Asynchronous Provisioning to Fictitious Dormitory Application via ActiveMQ

Schema of the Example



Evolveum

Implementation: Resource Definition

```
<name>Dormitory Test (async)</name>
<connectorRef type="ConnectorType">
    <filter>
        <q:equal>
            <q:path>connectorType</q:path>
            <q:value>AsyncProvisioningConnector</q:value>
        </q:equal>
    </filter>
</connectorRef>
<connectorConfiguration>
    <conf:targets>
        <jms>
            <connectionFactory>localhostConnectionFactory</connectionFactory>
            <username>admin</username>
            <password>admin123</password>
            <destination>TestQueue</destination>
        </jms>
    </conf:targets>
    <conf:predefinedTransformation>simplifiedJson</conf:predefinedTransformation>
</connectorConfiguration>

<capabilities>
    <configured xmlns:cap="http://midpoint.evolveum.com/xml/ns/public/resource/capabil
        <cap:read>
            <cap:cachingOnly>true</cap:cachingOnly>
        </cap:read>
    </configured>
</capabilities>
```

(1) connector type

(2) where to send messages to

(3) physical message format

(1) attribute caching

```
<xsd:sequence>
    <xsd:element name="login" type="xsd:string" minOccurs="0"/>
    <xsd:element name="givenName" type="xsd:string" minOccurs="0"/>
    <xsd:element name="familyName" type="xsd:string" minOccurs="0"/>
    <xsd:element name="email" type="xsd:string" minOccurs="0"/>
    <xsd:element name="faculty" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
```

(3) logical message format (schema)

```
{
  "operation" : "add",
  "objectClass" : "AccountObjectClass",
  "attributes" : {
java.naming.factory.initial=org.apache.activemq.artemis.jndi.ActiveMQInitialContextFactory
connectionFactory.localhostConnectionFactory=tcp://localhost:61616
queue.TestQueue=TestQueue
queue.DormitoryRequestsQueue=DormitoryRequestsQueue
  }
}
```

jndi.properties wrapped in a JAR on classpath

```
{
  "operation" : "modify",
  "objectClass" : "AccountObjectClass",
  "primaryIdentifiers" : {
    "login" : [ "banderson" ]
  },
  "secondaryIdentifiers" : { },
  "changes" : {
    "givenName" : {
      "replace" : [ "BOB" ]
    }
  }
}
```

```
{
  "operation" : "delete",
  "objectClass" : "AccountObjectClass",
  "primaryIdentifiers" : {
    "login" : [ "banderson" ]
  },
  "secondaryIdentifiers" : { }
}
```


Evolveum

https://docs.evolveum.com/midpoint/reference/resources/asynchronous/outbound/configuration/#resource-definition

Custom Request Message Transformation

```
        <conf:transformExpression>
            <script>
                <language>http://midpoint.evolveum.com/xml/ns/public/expression/language#velocity</language>
                <code>#set ( $request = $requestFormatter.changeMapAsAttributes().identifiersAsAttributes().createRequest() )
#set ( $attrs = $request.attributesSimplified )
#if ($request.isDelete())
delete
$!attrs["login"]
#else
addOrUpdate
$!attrs["login"]
gn: $!attrs["givenName"]
fn: $!attrs["familyName"]
m: $!attrs["email"]
f: $!faculty"]
#end</code>
            </script>
        </conf:transformExpression>
```

```
addOrUpdate
banderson
gn: Bob
fn: Anderson
m: banderson@example.edu
f: Faculty of Mathematics, Physics and Informatics
```

(1) account addition

```
addOrUpdate
banderson
gn: BOB
fn: Anderson
m: banderson@example.edu
f: Faculty of Mathematics, Physics and Informatics
```

(2) account modification

```
delete
banderson
```

(3) account deletion

https://docs.evolveum.com/midpoint/reference/resources/asynchronous/outbound/configuration/#using-custom-transformation

Evolveum

Live Demo

- d:\mp-home\apache-artemis-2.17.0\bin\artemis.cmd create d:\tmp\broker --user admin --password admin123 --require-login
- paste queue definitions to d:\tmp\broker\etc\broker.xml (addresses)
- d:\tmp\broker\bin\artemis run
- git clone https://github.com/mederly/webinar-async-provisioning-2021-05
- mvn clean install
- mvn exec:java
- mkdir d:\tmp\midpoint-4.3.1\var\lib
- copy d:\tmp\webinar-async-provisioning-2021-05\target*jndi.jar d:\tmp\midpoint-4.3.1\var\lib
- d:\tmp\midpoint-4.3.1\bin\start

https://github.com/mederly/webinar-async-provisioning-2021-05

Summary

- Asynchronous resources are experimentally supported (both ways)
- Main reasons to use:
  - When there is no direct access (source & target)
  - When we need immediate updates from a source
  - When a provisioning target is too slow (in the future)
- Main things to consider:
  - No direct access → no guarantees of long-term consistency
  - Async provisioning → no direct feedback (limited/slower adaptation to the real state)
  - Higher complexity
  - The current support is experimental

Feel free to experiment, and let us know about your experiences.

Evolveum

Discussion

Questions?

Comments?

For more information please visit
https://docs.evolveum.com/midpoint/reference/resources/asynchronous/outbound/

Evolveum

Thank you for your attention

If any questions occur, feel free to ask at **sales@evolveum.com**

Also **follow us** on our social media for further information!

 /Evolveum

 /Evolveum

 /Evolveum

 @Evolveum

 /Evolveum

Evolveum