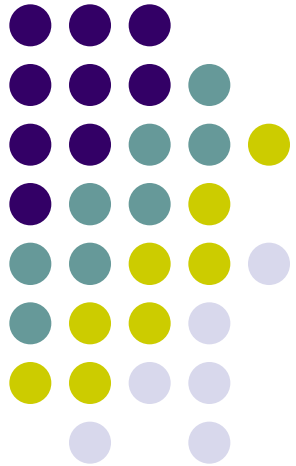
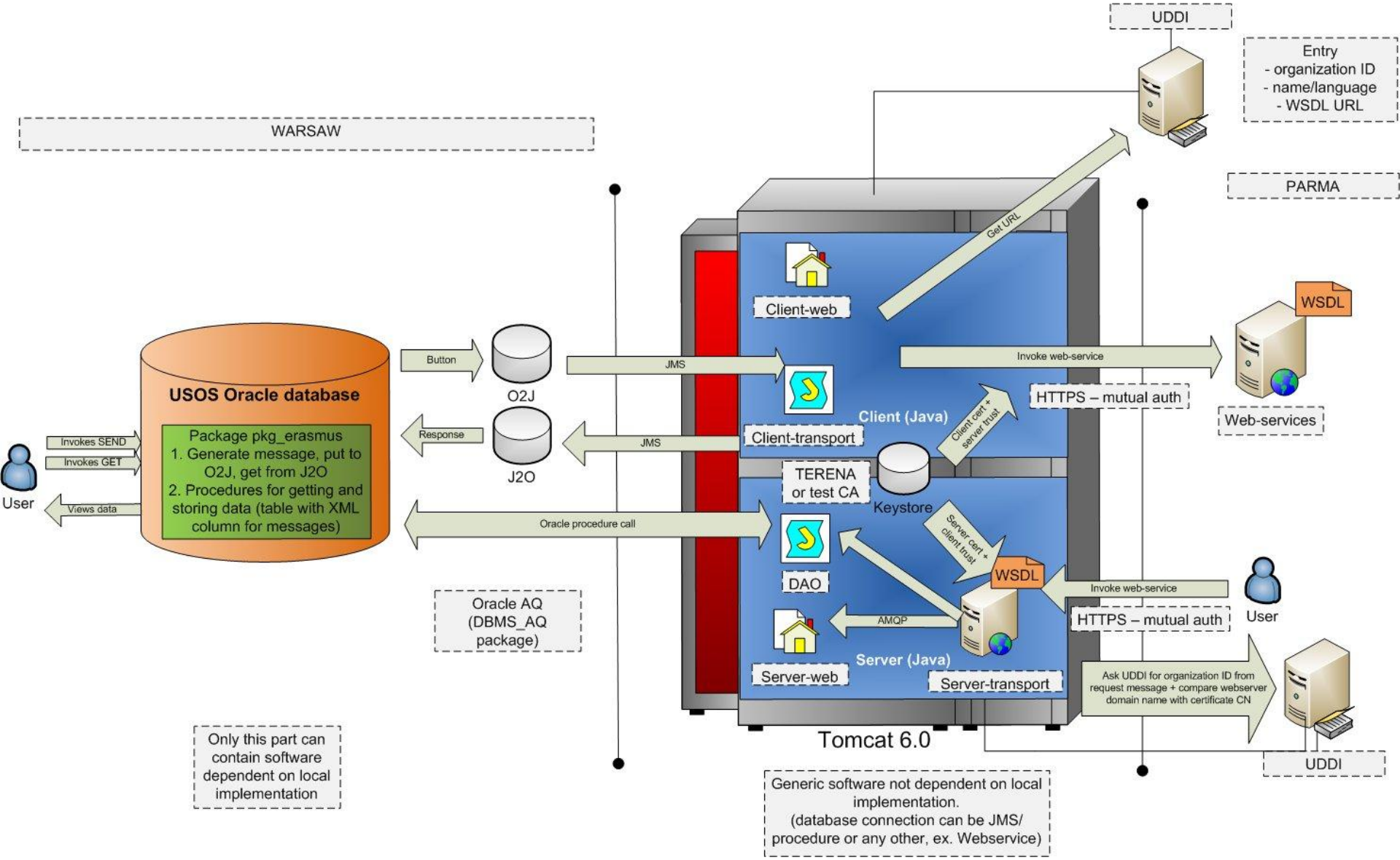
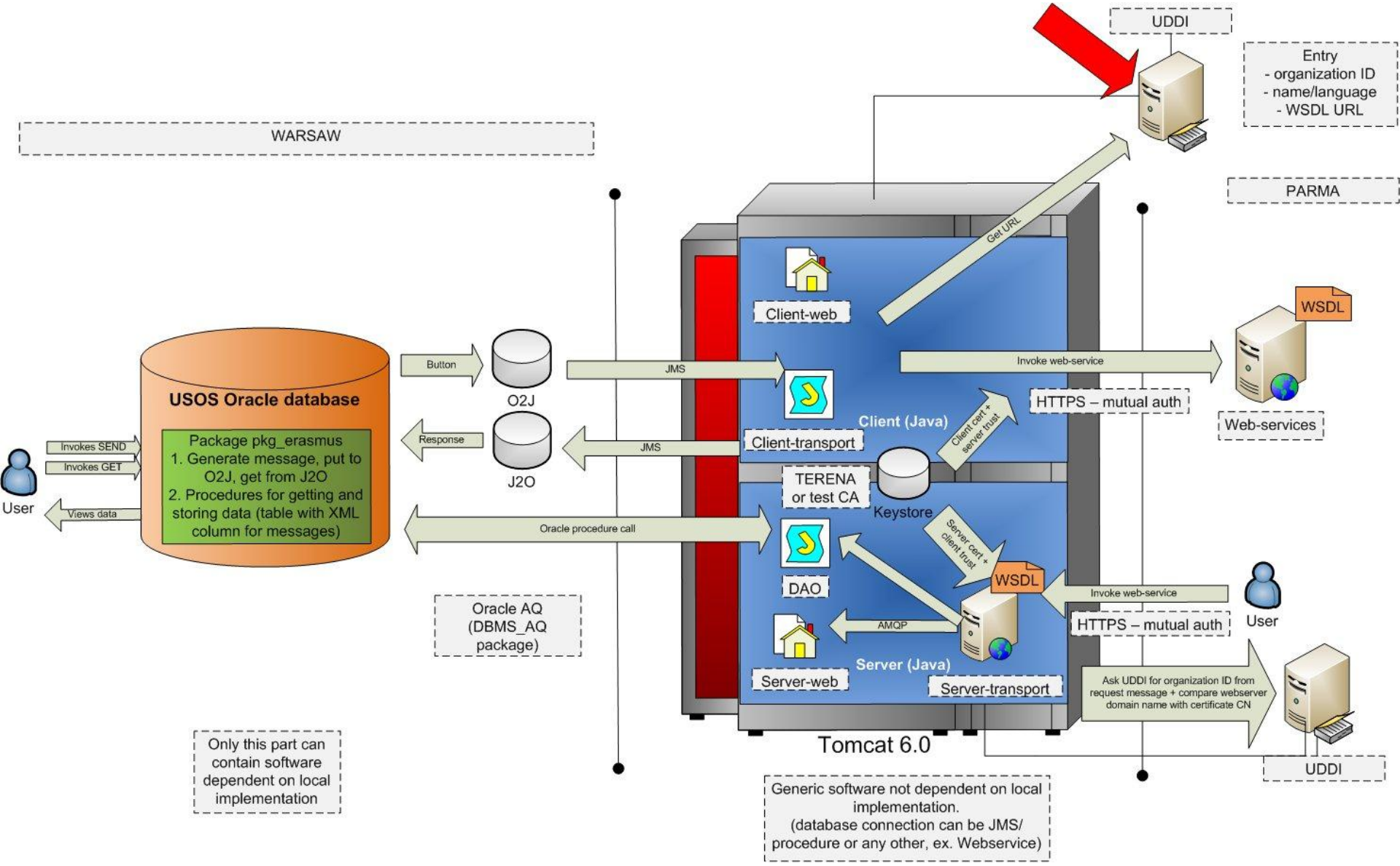
The Mobility Project

Technical Architecture

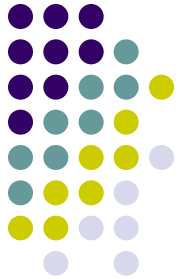


Michał Kurzydłowski
University of Warsaw
michalk@mimuw.edu.pl

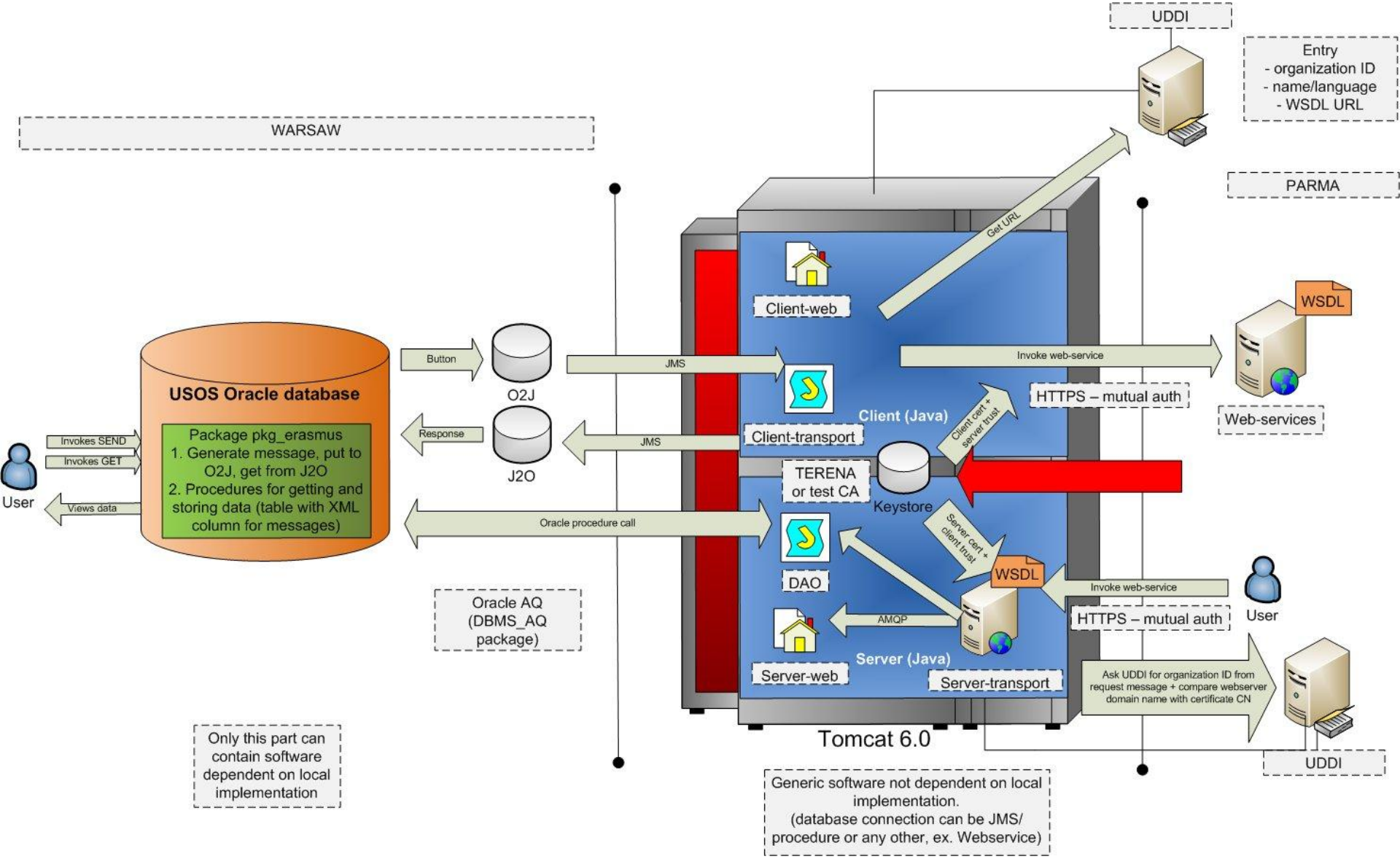

UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Button
JMS
O2J
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Client (Java)
Client-transport
Client cert + server trust
Response
JMS
J2O
TERENA or test CA
Keystore
Invokes SEND
Invokes GET
User
Views data
Oracle procedure call
Server cert + client trust
DAO
WSDL
Invoke web-service
HTTPS – mutual auth
User
Oracle AQ (DBMS_AQ package)
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Tomcat 6.0
UDDI
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)


UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Client (Java)
Client-transport
Client cert + server trust
TERENA or test CA
Keystore
Server cert + client trust
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Button
O2J
JMS
Response
J2O
Invokes SEND
Invokes GET
Views data
User
Oracle procedure call
DAO
WSDL
Invoke web-service
HTTPS – mutual auth
User
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
UDDI
Oracle AQ (DBMS_AQ package)
Tomcat 6.0
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

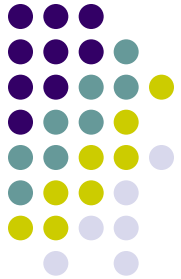
UDDI registry



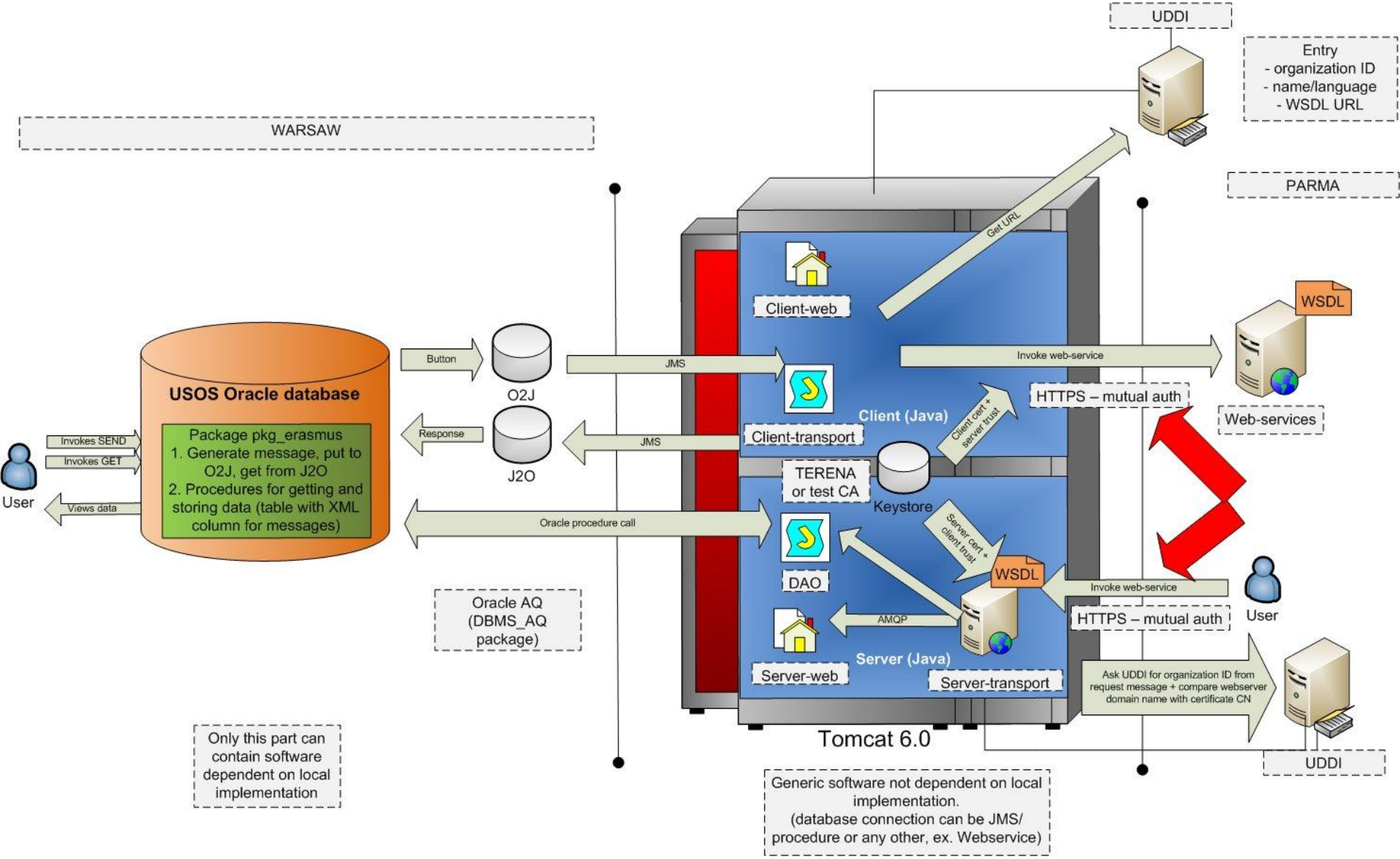
- Universal Description Discovery and Integration
  - Registry entry for every organization
    - Organization ID (eg. uw.edu.pl) – searching
    - Name in specified languages
    - Webservice location (WSDL URL) – domain name
  - Registry can be managed by client web app.
  - Many databases supported (eg. HSQLDB)
  - Technology: JUDDI 2.0 rc5 (opensource)
  - Important part in security policy (client/server trust)


UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Button
JMS
O2J
Client (Java)
Client-transport
Client cert + server trust
Response
JMS
J2O
TERENA or test CA
Keystore
Invokes SEND
Invokes GET
User
Views data
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Oracle procedure call
Server cert + client trust
DAO
WSDL
Invoke web-service
User
HTTPS – mutual auth
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Tomcat 6.0
UDDI
Oracle AQ (DBMS_AQ package)
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

Java KeyStore (JKS)

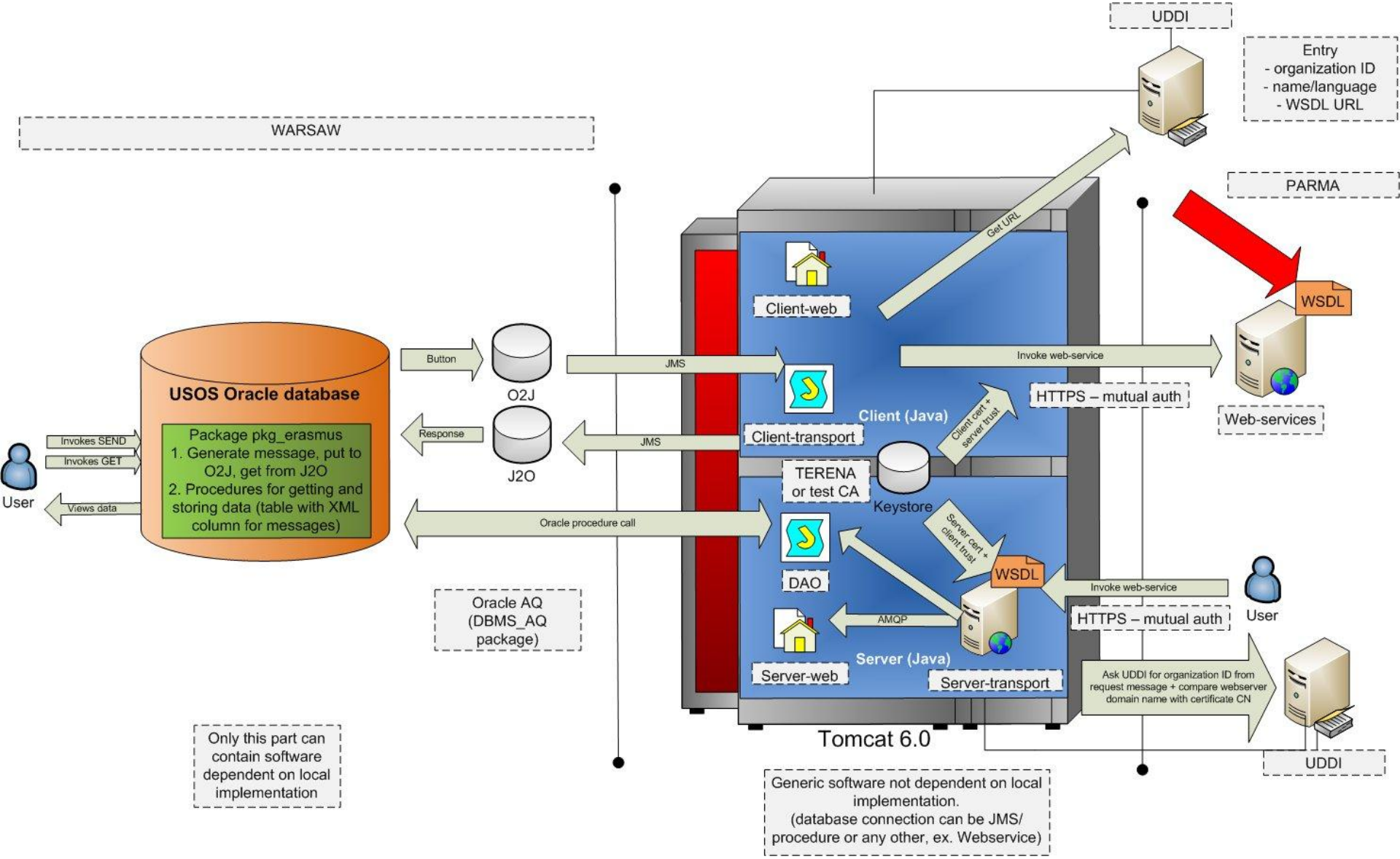


- Certificate signed by Terena SSL CA or Mobility test CA
  - Server and client certificate
  - CN must correspond to webserver domain name (WSDL URL in partner or global UDDI)
- Terena SSL CA for client/server certificate trust restriction
- Mobility test CA
- Used by client (web or database invocation) and server (webservice traffic must be https)


UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Client (Java)
Client-transport
Client cert + server trust
TERENA or test CA
Keystore
Button
O2J
JMS
Response
J2O
JMS
Invokes SEND
Invokes GET
User
Views data
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Oracle procedure call
DAO
Server cert + client trust
WSDL
Invoke web-service
HTTPS – mutual auth
User
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Tomcat 6.0
UDDI
Oracle AQ (DBMS_AQ package)
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

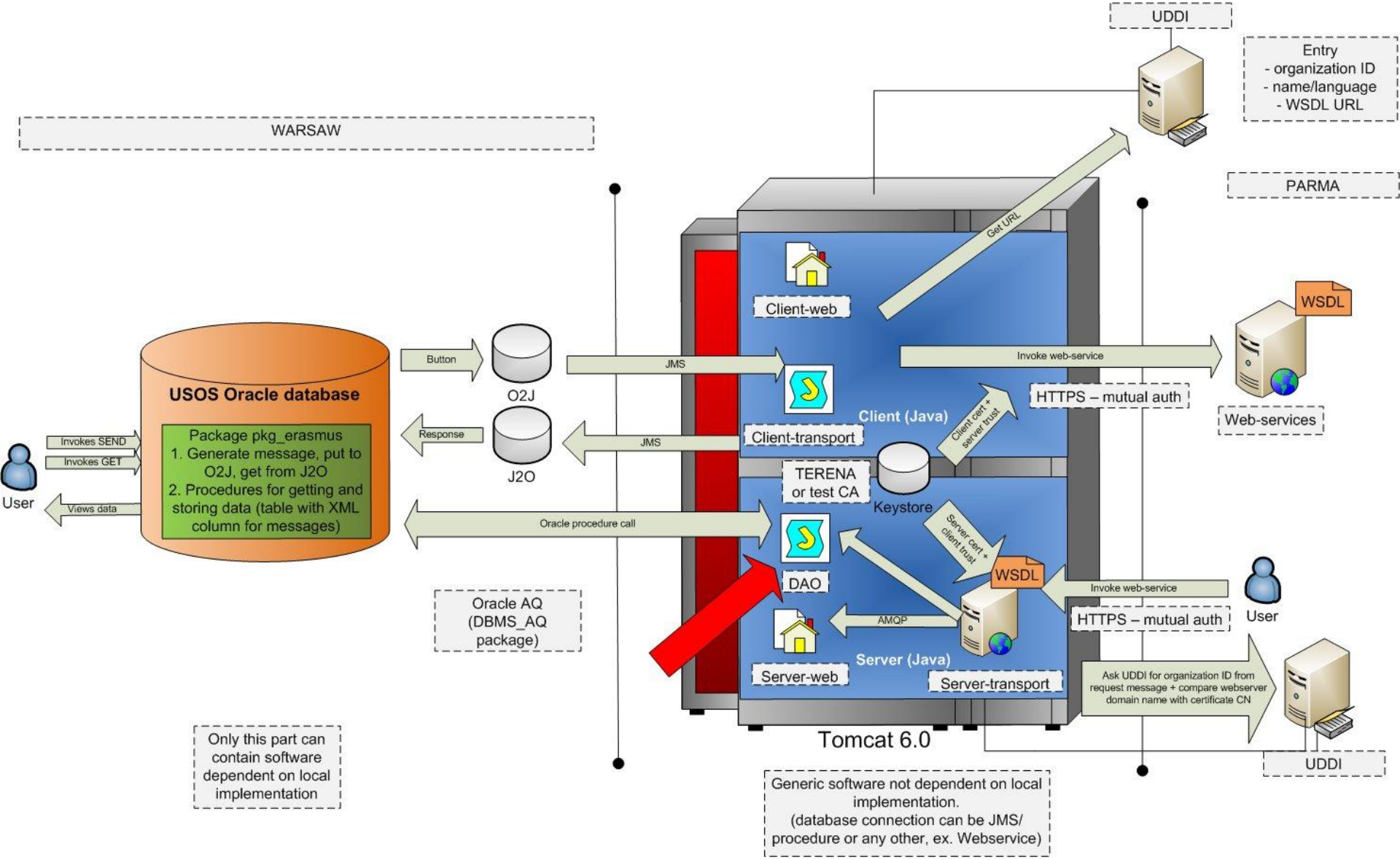
TLS mutual authorization

- Client request:
  - Find the WSDL URL of the partner mobility node in UDDI registry (by organization ID or name).
  - Do a TLS connection to server (the certificate DN must correspond to the domain name of the server - this is checked by the standard library)
- Server response:
  - Get the client certificate
  - Get the client organization ID from the request message (for some methods verification isn't needed, eg. getOrganizationData). Get the WSDL URL of the mobility node for this organization ID from the UDDI registry. Compare the domain name of the mobility node with the client certificate CN.

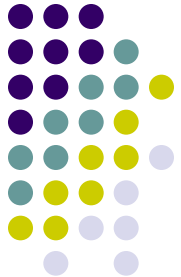

UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Button
O2J
JMS
Client (Java)
Client-transport
Client cert + server trust
Response
J2O
JMS
TERENA or test CA
Keystore
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Invokes SEND
Invokes GET
Views data
User
Oracle procedure call
DAO
Server cert + client trust
WSDL
Invoke web-service
HTTPS – mutual auth
User
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Oracle AQ (DBMS_AQ package)
Tomcat 6.0
UDDI
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

WSDL v2.1

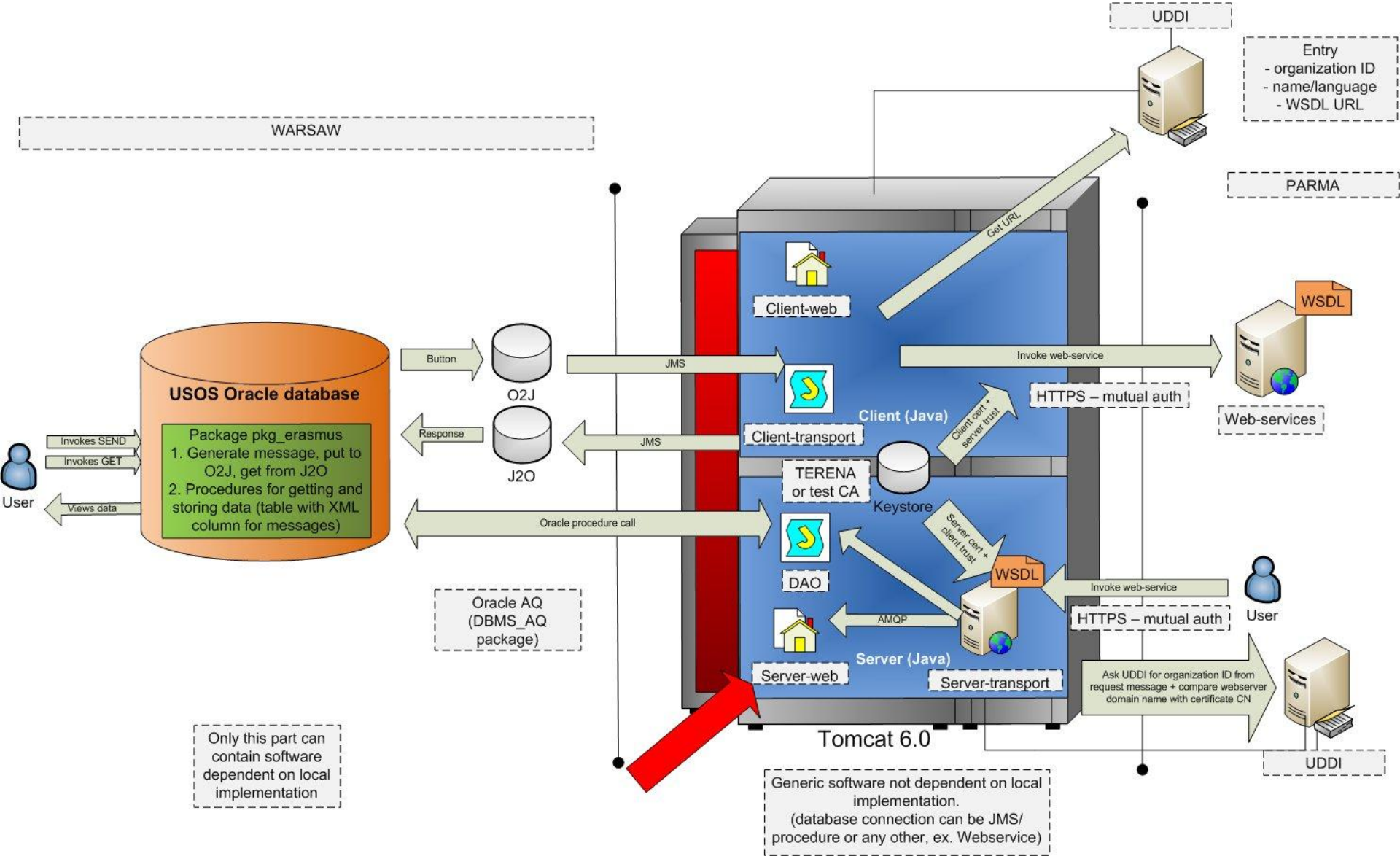
- Changes concerning personal data
  - http://wiki.teria.no/display/RS3G/Person
- Only sample XML data files and database message generation procedures needed correction
- Webservice recompilation necessary for message validation
- Other application parts do not rely on fields


UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Button
JMS
O2J
Client (Java)
Client-transport
Client cert + server trust
Response
JMS
J2O
TERENA or test CA
Keystore
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Invokes SEND
Invokes GET
Views data
User
Oracle procedure call
Server cert + client trust
DAO
WSDL
Invoke web-service
HTTPS – mutual auth
User
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Oracle AQ (DBMS_AQ package)
Tomcat 6.0
UDDI
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

WebService Server (DAO)

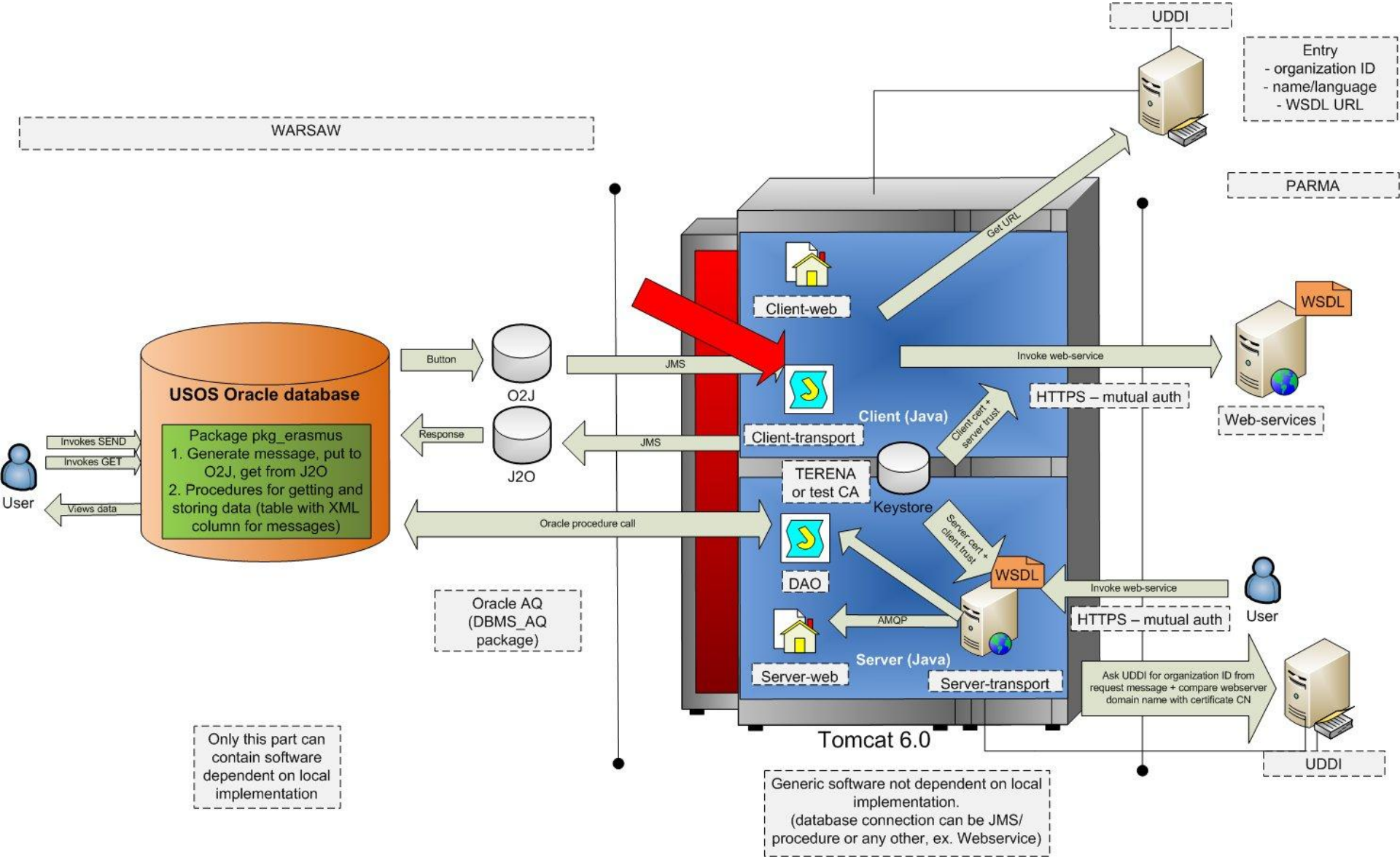


- Every method corresponds to a java bean that enables access to required data (data access object) for preparing response or storing data specific to the chosen method
  - There is only one API for DAO beans
  - The implementation can therefore be added by writing new bean and updating the configuration XML file (DAO beans are injected using Spring)
  - The basic DAO loads data from XML files located in specified folder tree (simulates a database)
  - DatabaseDAO executes Oracle procedures (get/send)

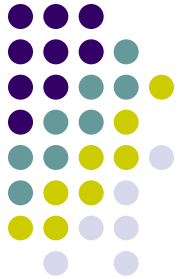

UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Button
JMS
O2J
Client (Java)
Client-transport
Client cert + server trust
TERENA or test CA
Keystore
Response
JMS
J2O
Invokes SEND
Invokes GET
User
Views data
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Oracle procedure call
DAO
Server cert + client trust
WSDL
Invoke web-service
User
HTTPS – mutual auth
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Tomcat 6.0
UDDI
Oracle AQ (DBMS_AQ package)
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

Server Web (Log)

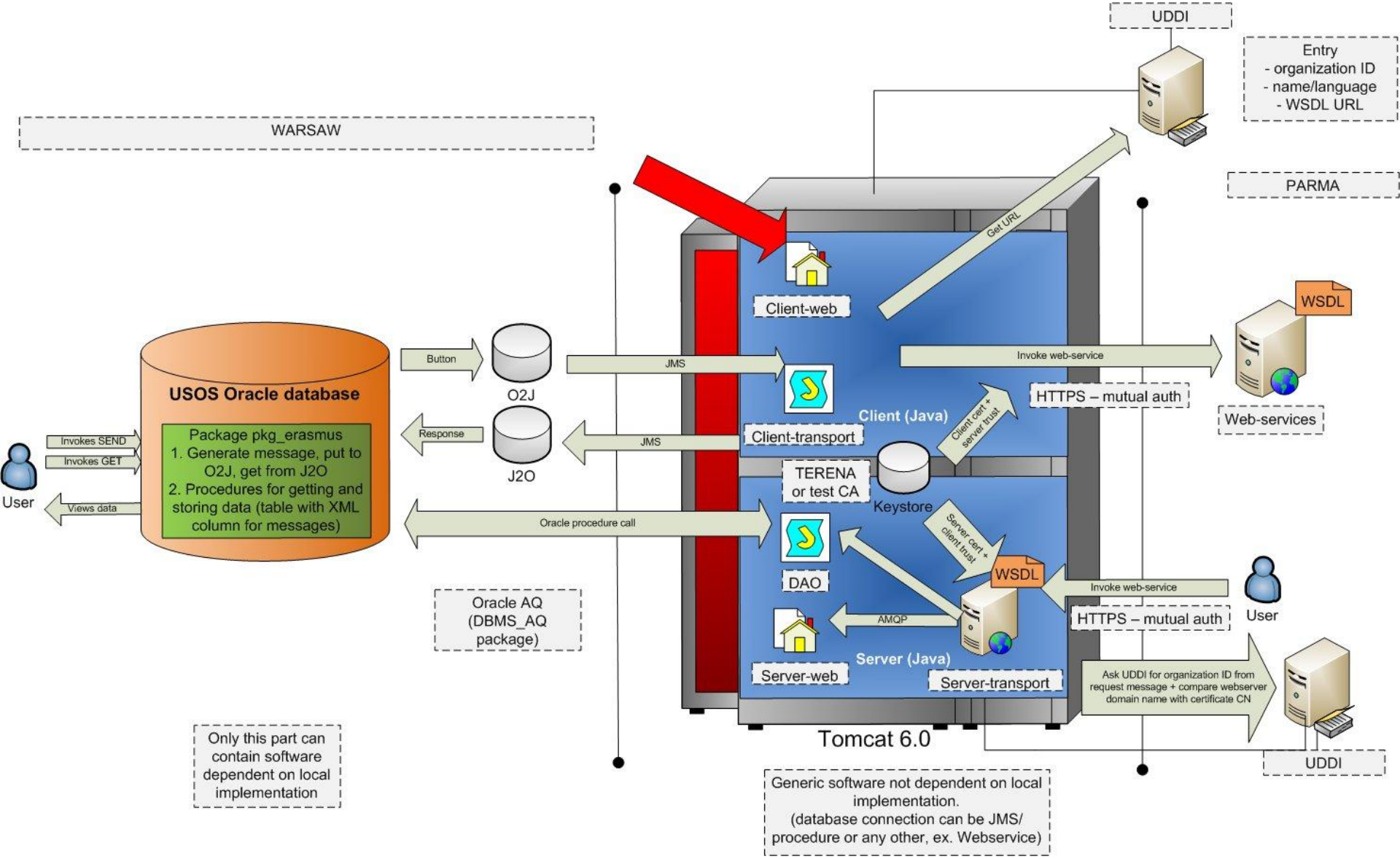
- All received data is logged and stored
- Data is stored in an AMQP queue
  - Started separately (Apache Qpid broker)
- The data logged in the queue can be viewed on a server web page


UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Button
O2J
JMS
Client (Java)
Client-transport
Client cert + server trust
Response
J2O
JMS
TERENA or test CA
Keystore
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Invokes SEND
Invokes GET
Views data
User
Oracle procedure call
DAO
Server cert + client trust
WSDL
Invoke web-service
HTTPS – mutual auth
User
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
Tomcat 6.0
UDDI
Oracle AQ (DBMS_AQ package)
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

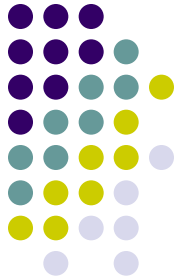
Client transport (JMS)



- JMS listener waits for request message sent by an Oracle procedure (activated by a „button” in Oracle forms)
- Message consists of SOAP request and a receiver organization ID (JMS property)
- UDDI lookup for WSDL URL of the receiver
- Webservice invocation (HTTPS)
- Put SOAP response (can be error) in the response queue (the same correlation ID)


UDDI
Entry
- organization ID
- name/language
- WSDL URL
WARSAW
PARMA
Get URL
Client-web
WSDL
Invoke web-service
HTTPS – mutual auth
Web-services
Client (Java)
Client-transport
Client cert + server trust
TERENA or test CA
Keystore
Button
O2J
JMS
Response
J2O
USOS Oracle database
Package pkg_erasmus
1. Generate message, put to O2J, get from J2O
2. Procedures for getting and storing data (table with XML column for messages)
Invokes SEND
Invokes GET
Views data
User
Oracle procedure call
DAO
Server cert + client trust
WSDL
Invoke web-service
HTTPS – mutual auth
User
AMQP
Server (Java)
Server-web
Server-transport
Ask UDDI for organization ID from request message + compare webserver domain name with certificate CN
UDDI
Oracle AQ (DBMS_AQ package)
Tomcat 6.0
Only this part can contain software dependent on local implementation
Generic software not dependent on local implementation.
(database connection can be JMS/ procedure or any other, ex. Webservice)

Web Client



- JSF (Java Server Faces) web application
  - IceFaces and Ajax enable to write response data and error messages without refreshing the whole page
- Choose the appropriate method from a list (generated dynamically from WSDL) and the universities that act as client and server (UDDI)
- The input data is loaded from XML example files depending on the choice of both universities
  - It can than be modified and validated before being sent
- UDDI manager also available

Discussion

- Thank you for your attention
- Any questions?