

OLTM PROOF OF CONCEPT

Naor Toledo Pinto
May 2014

OLTM PROOF OF CONCEPT

OLTM Proof Of Concept

Overview 

Requirements

Design & Implementation

OVERVIEW

- THE PURPOSE OF THIS DOCUMENT IS TO DISCUSS AT A HIGH LEVEL THE DESIGN OF THE OLTM PROOF OF CONCEPT IMPLEMENTATION.

REQUIREMENTS



1- APPLICATIONS PROOF OF CONCEPT

- THIS IMPLEMENTATION SHOULD DEMONSTRATE THE BASICS OF AN "IN MEMORY" TRANSLATION SERVICE. THE MAIN "SERVER" APPLICATION SHOULD PROVIDE TRANSLATION SERVICES.
- THE APPLICATION WILL HOLD THE "DICTIONARY" AND WILL PROVIDE THE RESOLUTION FOR ANY OF ENTRIES SAVED IN MEMORY.
- THE CLIENT APPLICATION SHOULD USE SOCKET TECHNOLOGY TO INTERACT WITH THE TRANSLATION SERVICE.
- THE SERVICE SHOULD EXPOSE THE MAIN LIFE CYCLE OPERATIONS FOR THE MAINTENANCE OF THE TRANSLATION DICTIONARY: CREATE, READ, UPDATE, DELETE AND A PEEK TASK TO INFORM IF A PARTICULAR ENTRY IS PART OF THE COLLECTION.

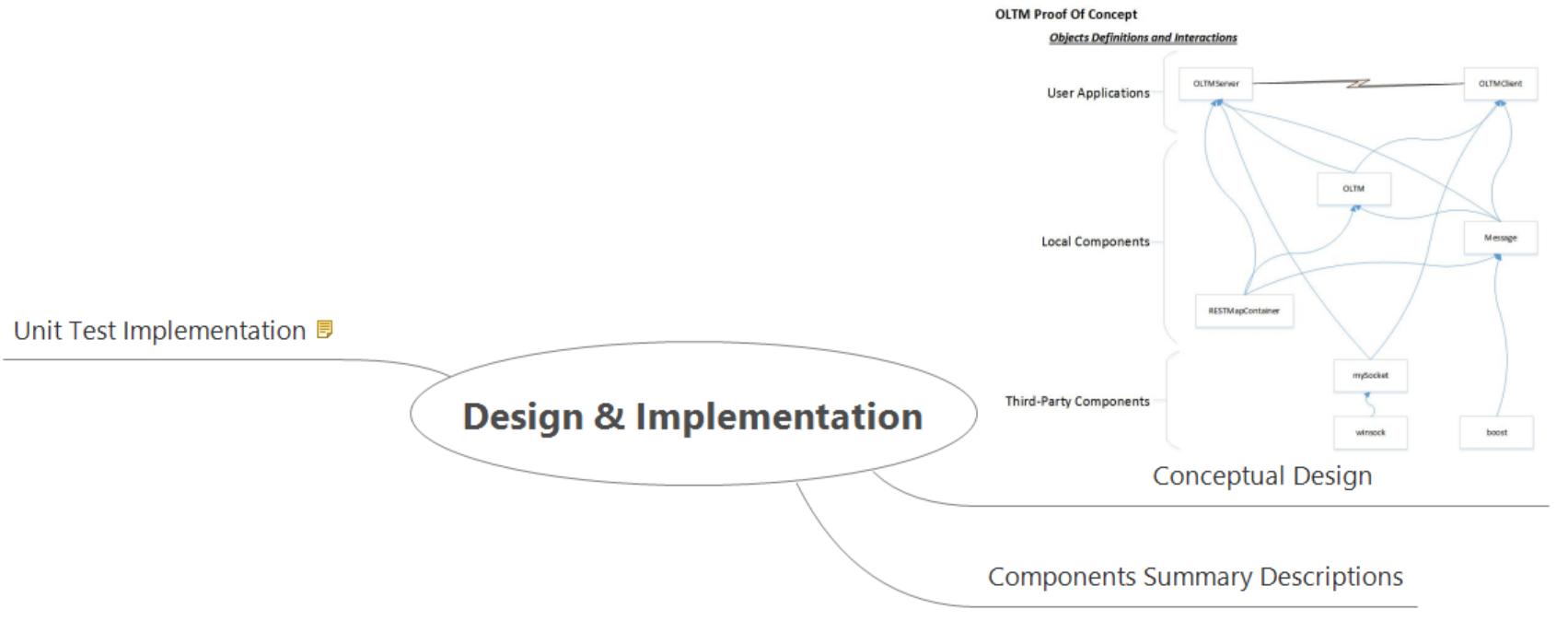
2- DESIGN FOCUS

- THE FOCUS OF THIS IMPLEMENTATION IS DEMONSTRATE THE BASICS OF THE FUNCTIONALITY AND SOME GOOD CODING PRACTICES.
- THIS IMPLEMENTATION WILL NOT STRIVE TO DEAL WITH ASPECTS OF SCALABILITY AND PERFORMANCE.

3- UNIT TEST DEMONSTRATION

- IN ADDITION, THIS IMPLEMENTATION WILL DEMONSTRATE A LITTLE BIT ABOUT UNIT TESTING.
- THE UNIT TESTING WILL DEMONSTRATE THE BASICS OF WHAT CAN BE DONE WITH UNIT TEST, HOWEVER IT WILL COVER ONLY THE BASIC OPERATIONS AND MAIN COMPONENTS CREATED FOR THIS SOLUTION.

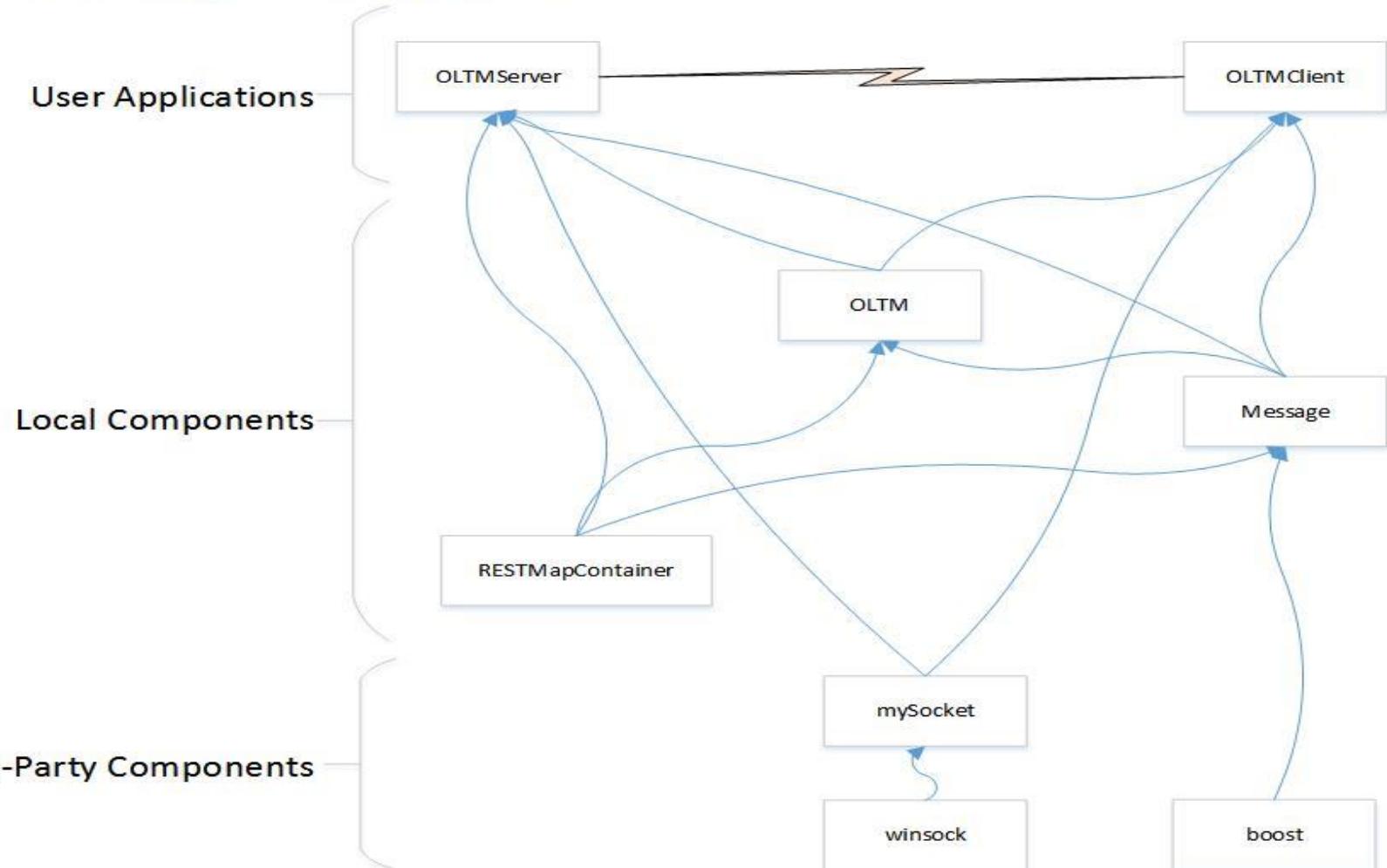
DESIGN & IMPLEMENTATION



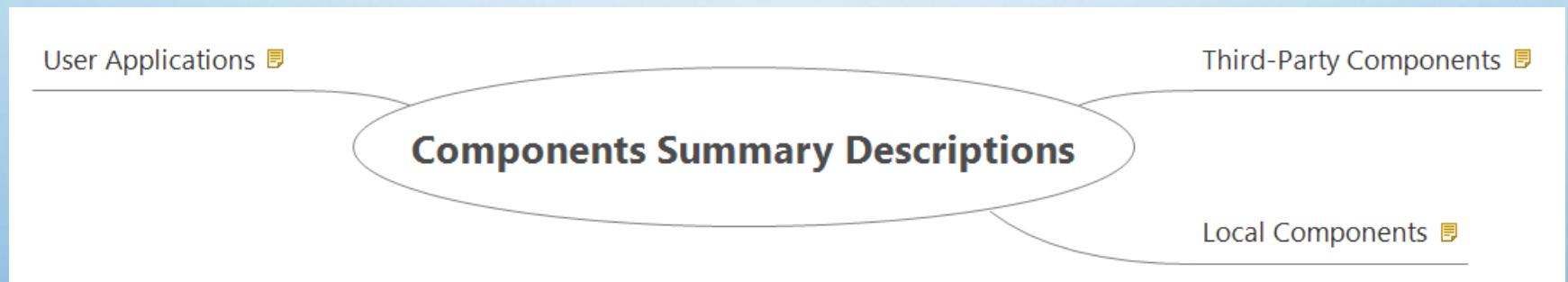
CONCEPTUAL DESIGN

OLTM Proof Of Concept

Objects Definitions and Interactions



COMPONENTS SUMMARY DESCRIPTIONS



THIRD-PARTY COMPONENTS

boost 

Winsock 

Third-Party Components 

mySocket 

- BESIDES THE NORMAL SUPPORT OF THE LANGUAGE AND STANDARD LIBRARIES WE HAVE INCLUDED SOME LIBRARIES THAT ARE WORTH MENTIONING AS THE BASE FOR THIS IMPLEMENTATION.
- THIS IMPLEMENTATION WAS CODED AND TESTED IN WINDOWS BASED C++ AND COMPILED WITH MICROSOFT STUDIO 2008.

WINSOCK

- FOR CONNECTIVITY WE HAVE USED THE WINSOCK FLAVOR FOR SOCKET SUPPORT.

MY SOCKET

- INSTEAD OF WRITING THE IMPLEMENTATION USING THE STANDARD SOCKET API'S WE DECIDED TO USE A "WRAPPING" API CREATED BY LY YANG YU.
- IN ADDITION TO "WRAPPING" THE BASICS OF SOCKETS INTO SOME OO IMPLEMENTATION THIS API PROVIDES SOME LOGGING SERVICES.
- THIS API IS ONLY A "SAMPLE" IMPLEMENTATION- BUT IT HAS ENOUGH FOR THIS PROOF OF CONCEPT.

BOOST

- WE HAVE THE BOOST LIBRARY TO PROVIDE SUPPORT FOR TWO FEATURES: UNIT TESTING AND JSON.
- WE HAVE USED JSON AS THE SERIALIZATION FORMAT OF THE MESSAGES BETWEEN THE CLIENT AND SERVER APPLICATIONS.
- FOR THE UNIT TESTS WE HAVE THE USE THE MINIMAL_TEST IMPLEMENTATION.

LOCAL COMPONENTS



- AS BASE FOR THIS IMPLEMENTATION WE HAVE CREATED THREE CLASSES.
- TWO OF THE CLASSES ARE COMPLETELY GENERIC AND ONLY ONE ENCAPSULATES THE KNOWLEDGE ABOUT OLTMD.

RESTMAPCONTAINER

- THIS CLASS EXPOSES REST SERVICES TO MAINTAIN "IN MEMORY" MAP COLLECTIONS.
- THIS OBJECT IMPLEMENTS THE BASIC REST OPERATIONS: CREATE, READ, UPDATE AND DELETE. IN ADDITION WE IMPLEMENTED A "PEEK" OPERATION - FOR THIS WE HAVE REUSED THE HEAD METHOD.

MESSAGE

- THIS IS A GENERIC CLASS THAT DEALS WITH "MESSAGES."
- THE MAIN PURPOSE OF THIS OBJECT IS TO HOLD A MESSAGE CONTENT AND BEING ABLE TO PARSE IT INTO THEIR CORRESPONDENT FIELDS AND COMPOSE IT BACK AS A SINGLE CONTENT STREAM.
- AT THE MOMENT THIS OBJECT CAN HOLD MESSAGES FORMATTED AS JSON OR STANDARD STRINGS.
- THIS OBJECT COULD BE EXTENDED IN THE FUTURE TO SUPPORT OTHER FORMATS SUCH AS XML, CSV, ETC.

OLTM

- AS INDICATED BEFORE THIS IS THE ONLY CLASS THAT HAS ANY KNOWLEDGE ABOUT THE TRANSLATION DOMAIN.
- IT ENCAPSULATES ALL THE DOMAIN RELATED DEFINITIONS AND ITS MAIN OPERATIONS.
- THE OPERATIONS MIRROR THE STANDARD REST FUNCTIONS HOWEVER IT INCLUDE SOME ADDITIONAL ERROR HANDLING PERTINENT TO THE DOMAIN.
- THIS IS INTENDED TO BE USED ONLY AS A CLASS - THIS IS THE "BEHAVIOR" OF THE APPLICATION AND NONE OF ITS STATE.
- THE MAIN STATE OF THE APPLICATION WILL BE MANAGED WITH THE RESTMAPCONTAINER AND THE TRANSIENT STATE WITH THE MESSAGE OBJECT.

USER APPLICATIONS

User Applications 

OLTM Server 

OLTM Client 

- FOR THIS APPLICATION WE HAVE CREATED TWO APPLICATIONS.
- BOTH OF THOSE APPLICATIONS ARE COMMAND LINE DRIVEN.

OLTM SERVER

- THIS IS A STANDARD SERVER SOCKET APPLICATION.
- THIS APPLICATION WILL HOST A RESTMAPCONTAINER OBJECT - THIS COLLECTION IS THE HEART OF THE SYSTEM. IT WILL CONTAIN THE TRANSLATIONS.
- THE SOLE PURPOSE OF THE SERVER APPLICATION IS TO PARSE THE MESSAGES AND PROCESS THE REQUESTS AND SEND THE RESPONSES TO THE CLIENT APPLICATIONS.

OLTM CLIENT

- THIS IS A STANDARD SOCKET CLIENT APPLICATION.
- FOR EACH USE THE USER WILL PASS A COMMAND. THE APPLICATION WILL OPEN A SOCKET COMMUNICATION WITH THE SERVER APPLICATION AND REQUEST PROCESS FOR IT.
- THE COMMUNICATION PROTOCOL USES MESSAGE OBJECTS, AND FOR EACH INTERACTION THE CLIENT APPLICATION WILL OPEN A SOCKET SESSION AND WILL CLOSE IT AFTER THE SERVER SENDS A RESPONSE.
- THE PROTOCOL INCLUDES THE FOLLOWING SEQUENCE:
 - OPEN THE SOCKET COMMUNICATION
 - CREATE A "REQUEST" MESSAGE OBJECT
 - SERIALIZE IT AND SEND IT TO THE SERVER
 - AFTER THE PROCESS THE SERVER SENDS A "RESPONSE" MESSAGE
 - THE APPLICATION WILL PARSE THE RESPONSE AND INFORMS THE USER OF THE RESULTS
 - THE APPLICATION WILL COMMUNICATE THE SERVER TO CLOSE THE SOCKET SESSION

UNIT TEST IMPLEMENTATION

- WE HAVE USED THE MINIMAL_TEST FEATURE OF BOOST FOR THE UNIT TEST IMPLEMENTATION.
- THIS IS NOT A COMPREHENSIVE UNIT TEST.
- WE HAVE CREATED A SINGLE UNIT TEST APPLICATION FOR TESTING THE THREE MAIN CLASSES CREATED FOR THIS IMPLEMENTATION: RESTMAPCONTAINER, MESSAGE AND OLTM.
- THE CLIENT APPLICATION IS A TEST APPLICATION IN ITSELF - AS IT STANDS ONE CAN USE IT AS A MANUAL TEST FOR THE SERVER APPLICATION.
- IN THE SERVER APPLICATION ALL RELEVANT FUNCTIONS ARE PROVIDED BY THOSE THREE OBJECTS - ANYTHING ELSE IS PLUMBING AND ERROR HANDLING.