

# LOCALISATION PROJECT MANAGEMENT FRAMEWORK

**A Project By:**  
C-DAC GIST, Pune, India

**Presented by:**  
Kamal Pathak & Chandrakant D

# AGENDA

- ▶ Overview of Localization Framework
- ▶ Key Features/Benefits
- ▶ High Level Architecture of Localisation Framework
- ▶ Introduction & Design of Localization Project Management System
- ▶ Introduction & Design of Translation Memory Management System.

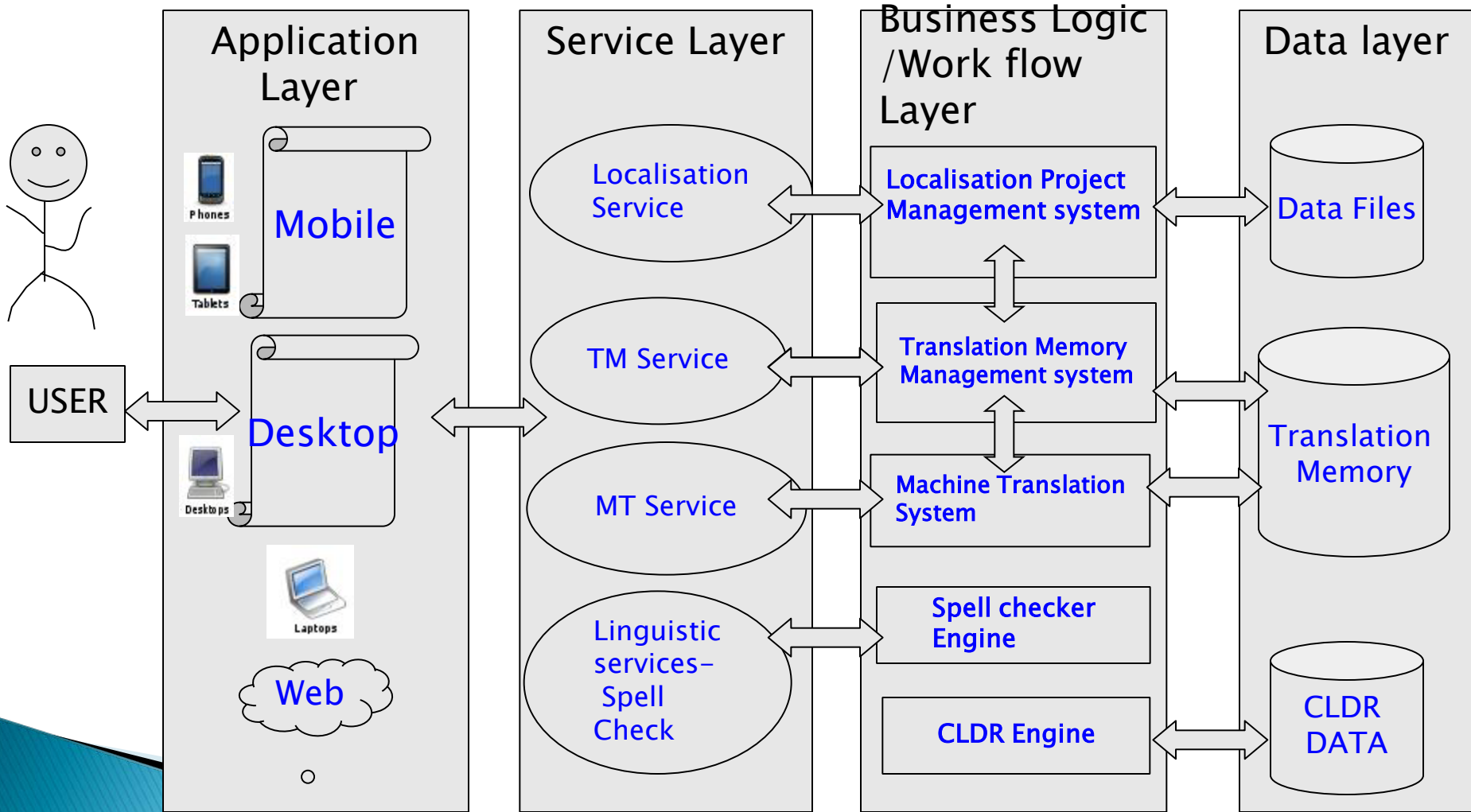
# Localisation Framework Overview

- ▶ Localisation framework will provide services to general user and Translation/Localisation community.
- ▶ Support of Management and automation of Localisation /Translation process.
- ▶ Support of Building and extraction of Translation memories.
- ▶ A common interface to various services such as Translation management, TM Management, Machine Translation, Spellcheckers etc.

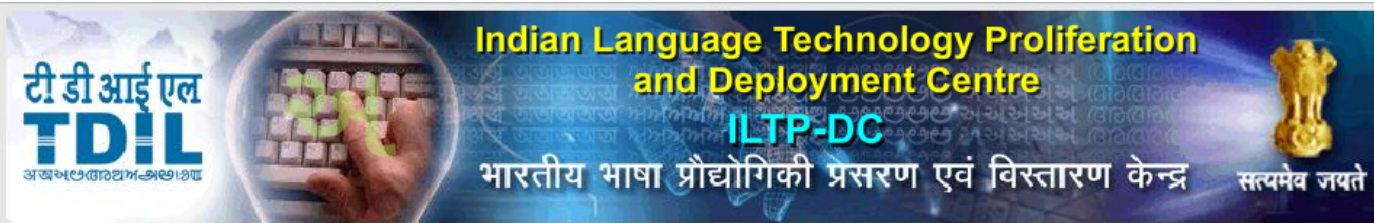
# Key Features of Localisation Framework

- ▶ Web Service support
- ▶ Extensible
- ▶ User Friendly
- ▶ A centralized repository for all the underlying services
- ▶ Adhered to guidelines & standards - such as W3C, XLIFF, XML, etc.

# High Level Architecture



# Establishment of Indian language Technology Proliferation & Deployment Centre



[Home](#) | [Download](#) | [Upload](#) | [Discussion Forums](#) | [Publications](#) | [Community](#) | [Archives](#) | [Live Chat](#) | [Feedback](#) | Search  [Go](#)  
[Success Stories](#) | [About Us](#) | [FAQ](#) | [Contact Us](#) | [Register here](#) | [Login](#)

- Standardization >
- Validators / Localization Tools >
- Linguistic Resources & Tools >
- Application Showcase >
- Research Areas >
- Technology Handshake >
- IPR >

## Standardization

[W3C](#)  
[Unicode](#)  
[Keyboard Standards](#)  
[Standardization Bodies](#)

<p><b>Machine Translation System</b></p> <p>Machine Translation Systems for Indian language to Indian Language and English to Indian Language are now available for public use. Fol...</p> <p style="text-align: right;"><a href="#">to access Click Here..</a></p>	<p><b>TDIL Programme</b></p> <p>The Department of Information Technology initiated the TDIL (Technology Development for Indian Languages) with the objective of developing Information Proc...</p> <p style="text-align: right;"><a href="#">more..</a></p>	<p><b>Free software tools and font CD</b></p> <p>Under the aegis of Department of Information Technology (DIT), Govt. of India, C-DAC GIST Pune has completed major initiative called National Rollout Plan...</p> <p style="text-align: right;"><a href="#">more..</a></p>	<p><b>Take a poll</b></p> <p>Does the new Rupee Symbol portray the uniqueness of India?</p> <p style="text-align: right;"><a href="#">more..</a></p>
---	---	---	--

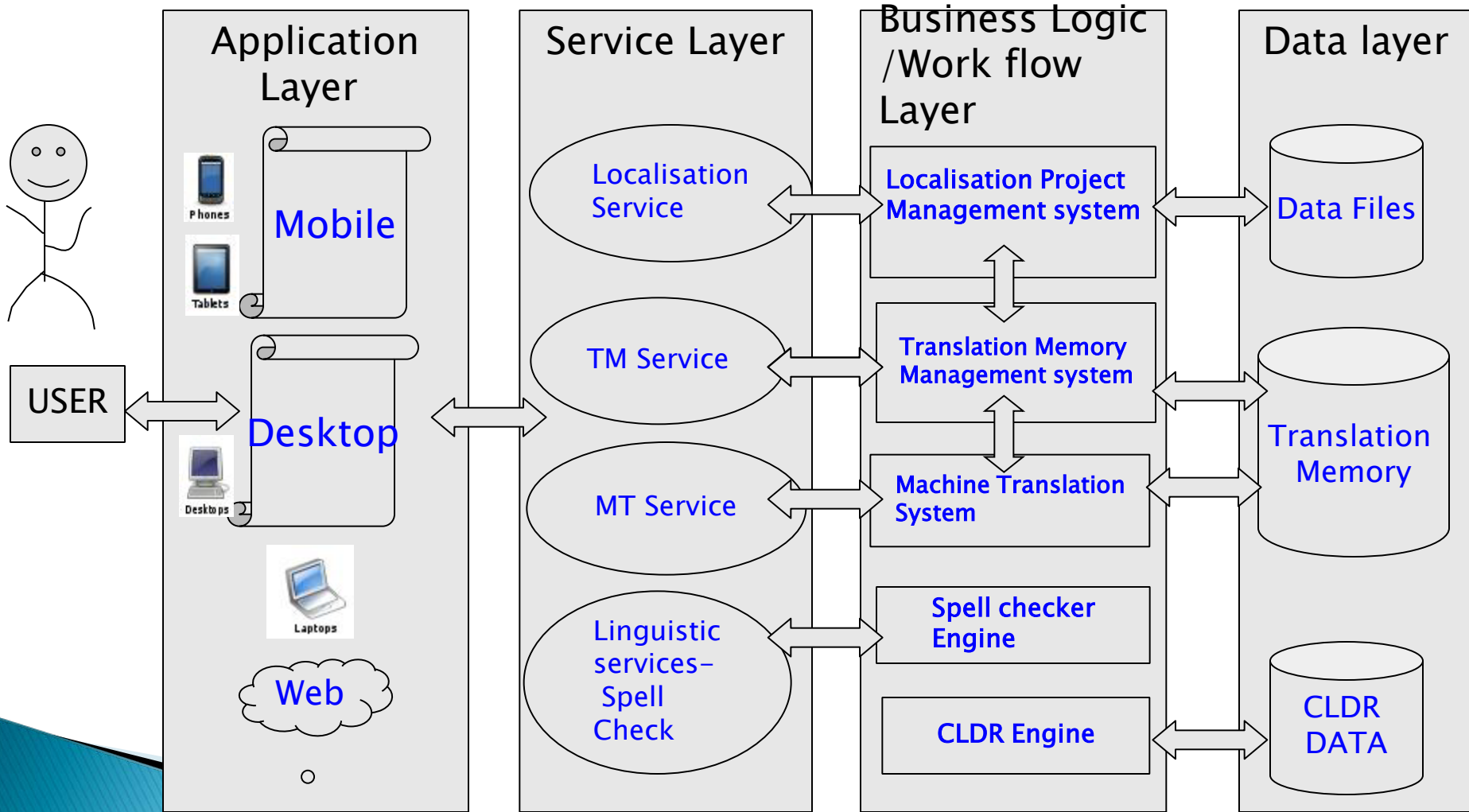
<p><b>New Events</b></p> <p>35th IUC, California, USA</p> <p><b>12 Sep 2011</b> RANLP 2011, Hissar, Bulgaria</p> <p><b>05 Sep 2011</b></p> <p style="text-align: right;"><a href="#">more...</a></p>	<p><b>Whats New at ILTP-DC !!</b></p> <div style="text-align: center;">  </div> <p><b>Documents and Tools</b></p> <p><a href="#">Script Grammar for Gujarati</a></p> <p><a href="#">CSS compliancy for North-Eastern scripts: A White Paper</a></p>	<p><b>Post Your Events</b></p> <p>Do you want forthcoming event to reach out to masses? Kindly post the same here by filling appropriate details.</p> <p style="text-align: right;"><a href="#">Post event...</a></p>
--	--	---

- Standardization
- Validators and Localization tools
- Linguistic Resources and Tools
- Application showcase
- Research areas

# Objectives of TDIL-DC

- Proliferation of Indian language technology and products.
- Research and Development of Technology, Software Tools and Applications for Indian Languages.
- Development of Standards for linguistic resources, tools and applications for interoperability
- Tools, technologies & other related resource under one roof.
- Baseline for Individuals, Industries & Academics for future work.

# High Level Architecture



# Localisation Project Management System

- ▶ Localisation Project Management System is a web based application for managing the translation projects with the help of online translation community.
- ▶ Required registration to use this service.

# Stake Holders in Localisation Project Management System

- ▶ Client representative
  - Creates and publish jobs.
  - Track progress
  - Download completed jobs.
- ▶ Project Manager
  - Configure job for Translator and Reviewer
  - Track progress
  - Managing Subscription for both clients and Translators/reviewers.
- ▶ Translator/Reviewer
  - Translation and review activities

# Overall Design Flow

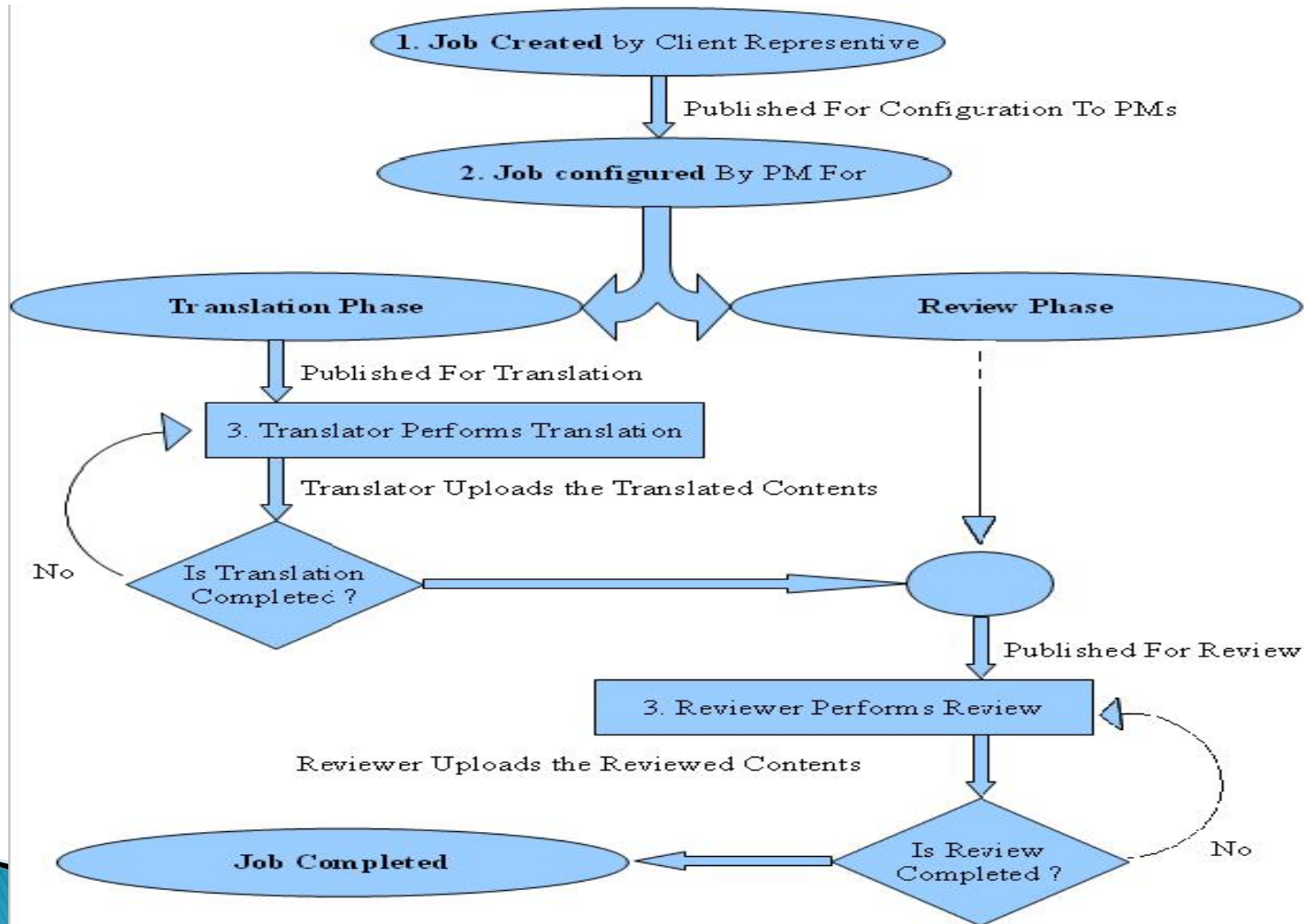
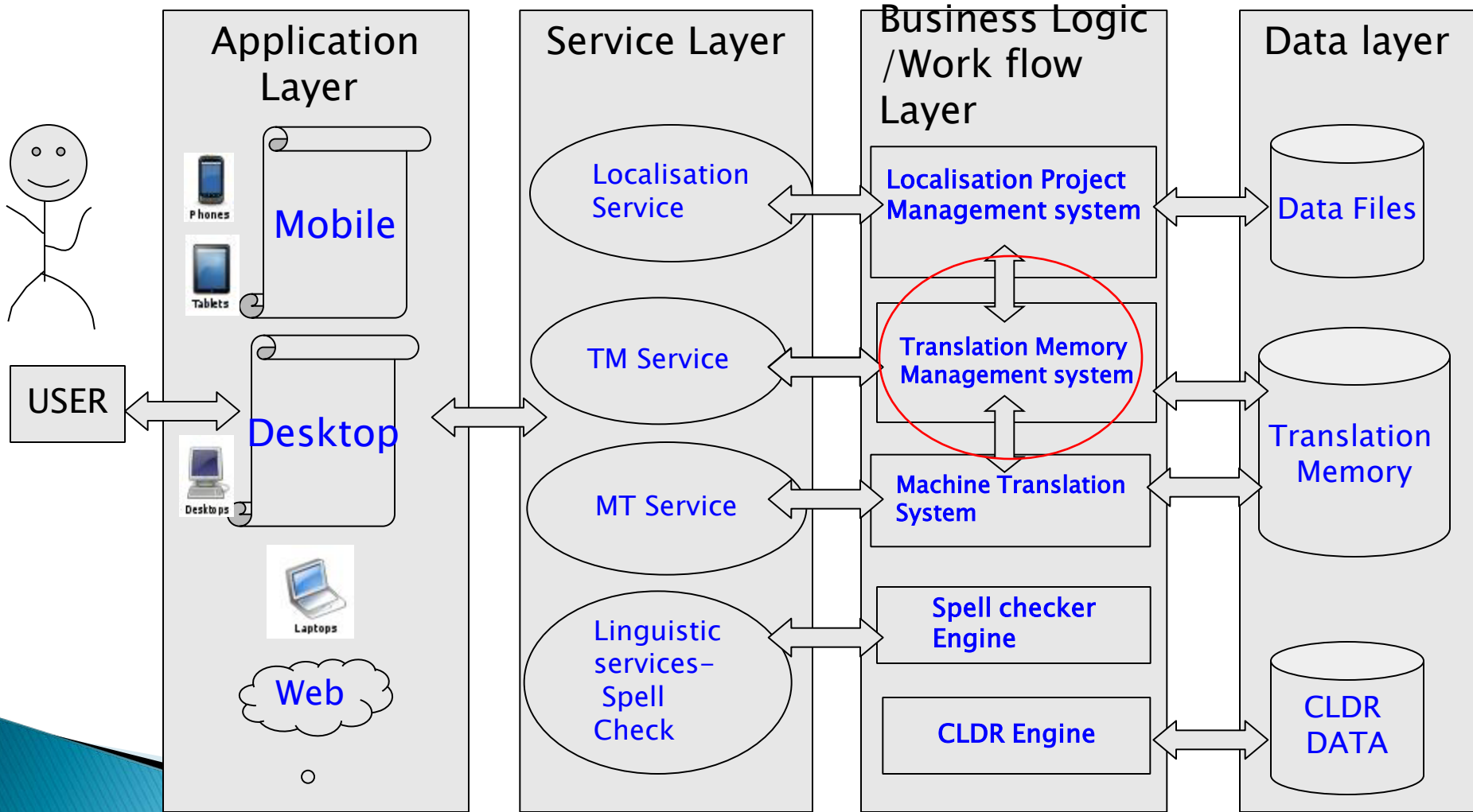


Figure: Community Job Processing Cycle

# High Level Architecture

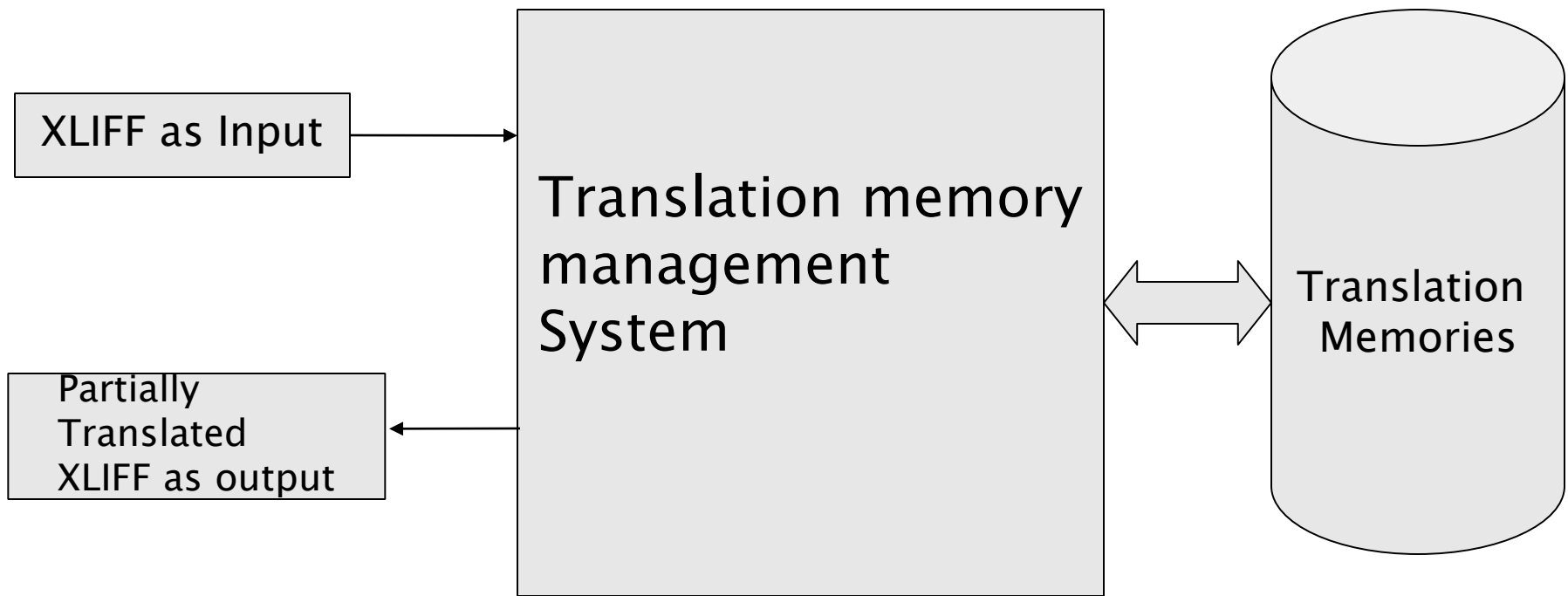


# Translation Memory Management System

## Scenario in C-DAC:

- ▶ CDAC receives documents for translation.
- ▶ Each text segment has to be manually translated.
- ▶ A huge database of translated segments is available.
- ▶ There was no mechanism to reuse this database.
- ▶ Cost of translation.

# TMM- Design



# Database Design

Also known as Translation Memory (T.M)

Source language(English) table

Segment_Id	BIGINT(10)
Source_Segment	VARCHAR(1000)

Target Language(Hindi/Marathi) table

Segment_Id	BIGINT(10)
Target_Segment	VARCHAR(1000)
Frequency	INT(2)

# Database design...Continued

## ► How are they linked?

The image shows two screenshots of the MySQL Query Browser interface. The left screenshot shows the query `SELECT * FROM english e;` and its result set. The right screenshot shows the query `SELECT * FROM hindi h;` and its result set. Red annotations link the English segment 'that command only works in chats not ims' (segment 4) to two corresponding Hindi translations (segments 4 and 5) in the Hindi result set.

**English Resultset 1:**

segid	seg
1	no such command
2	syntax error you typed the wrong number of arguments to t...
3	your command failed for an unknown reason
4	that command only works in chats not ims
5	that command doesn t work on this protocol

**Hindi Resultset 1:**

segid	seg	freq
1	ऐसा कोई कमांड नहीं है.	1
2	साम्यविन्यास त्रुटि: इस कमांड में आपने गलत संख्या का तर्क देकर दिया है.	1
3	आपका कमांड अज्ञात कारण से विफल हो गया.	1
4	इस कमांड केवल चैट्स में चलता है, आइएम में नहीं.	1
5	इस कमांड केवल आइएम में काम करता है., चैट्स में नहीं.	1

**Annotations:**

- segments stored with the same segid
- segment wid segid = 4 has 2 corresponding hindi translations

English segment pointing to corresponding alternate translations in Hindi.

# Database design...Continued

- ▶ Storing individual terms in a “terminology bank” :

## SourceTB

Term_Id	BIGINT(10)
Source_Term	VARCHAR(1000)

## TargetTB

Term_Id	BIGINT(10)
Target_Term	VARCHAR(1000)
Frequency	INT(2)

# Database design...Continued

## ▶ Index:

word	segids
abandon	711;15917;
abbrevi	12480;16580;
abc	6675;
abcd	19434;
abi	217;

## ▶ Key word search using Index

“**Ram** likes to **play hockey**”

Ram → 101; **105**; 1001

play → **105**; 1050; 1110

hockey → 1778; 1265; **105**

Therefore, the segment corresponding to segid 105 is the most relevant.

# Pre-translation Module

▶ **Aim:**

To retrieve probable translations in target language for a given English text segment.

▶ **Working:**

- *Scenario 1:* (segment is of one word)

Source segment: “hello”

Target Language: Hindi

Access the “Terminology Bank” and retrieve the corresponding target language term.

# Pre-translation Module...Continued

- *Scenario 2:* (source segment more than 1 word) Sentence

Source segment: “**Ram likes to play hockey**”

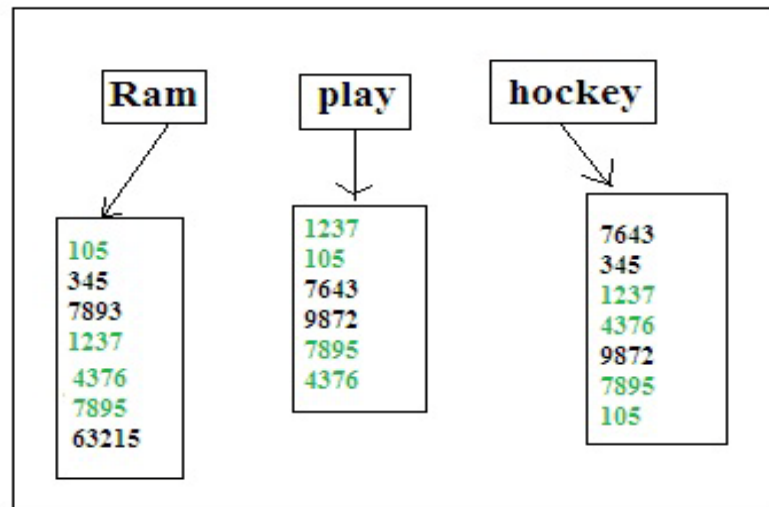
Target Language: Hindi

## Step 1: Remove Stopwords

result: Ram ----- -- play hockey

## Step 2: Stem keywords and search in Index

result:



# Pre-translation Module...Continued

- ▶ Intersection of the 3 sets gives those segids whose corresponding segments containing all the 3 keywords. (Highlighted in green)
- ▶ The possibly relevant segments are:
  - 105 : Ram likes to play hockey
  - 1237: Ram likes to play basketball and hockey
  - 4376: Ram does not like outdoor games like hockey and basketball.
  - 7895: Ram plays hockey.
- ▶ **Step 3: Levenshtein Distance**  
The Levenshtein's Edit distance of every segment in the above result set is calculated w.r.t. the source segment. ("**Ram likes to play hockey**" case.)

They are sorted in ascending order. The first 3 are considered.

Segid	Lev. Distance
105	0
1237	6
7895	11
4376	20

# Pre-translation module...Continued

## ▶ **Step 4:**

- Pull the corresponding target language translations.
- For each English segment obtained above, there can be more than one target language translations.
- The one with the highest value in “freq” column is selected.

## ▶ **Step 5:**

These are now displayed as the “probable suggestions”

# Translation Memory Building Module

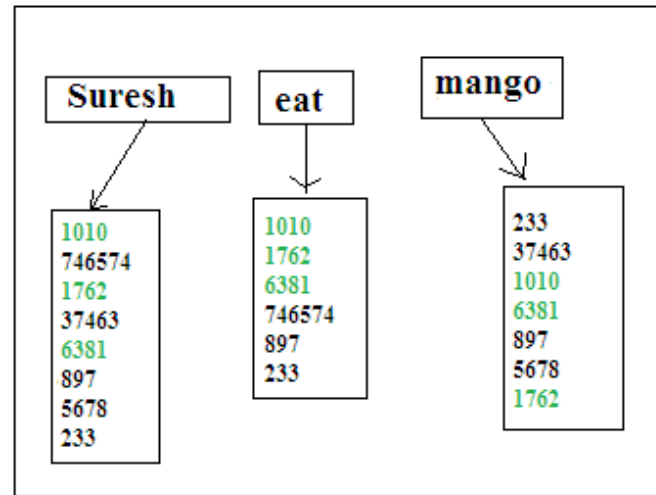
- ▶ **Aim:** To recognize newly entered translations typed in by the translator and to add them at appropriate positions in the Translation Memory.

## **Working:**

- ▶ *Scenario 1:* (segment is of one word)  
Source segment: “hello”  
Target Language: Hindi  
Dump the word in HindiTB.
- ▶ *Scenario 2:* (segment is of multiple words)  
Source segment: “**Suresh is eating a mango**”  
Target Language: Hindi
- ▶ The first 3 steps are same as those in pre-translation.

# TM Building Module.....continued

- ▶ Source segment: “**Suresh is eating a mango.**”
- ▶ The result of step 2:



- Those strings already present in the T.M and that are similar to the source string:
  - 1010: Suresh eats a mango.
  - 1762: Suresh eats mango ice-cream
  - 6381: Suresh is eating a fruit salad with mango slices in it.

# TM Building Module.....continued

- ▶ Source segment : “Suresh is eating a mango.”
- ▶ Step 3:
  - Calculate Levenshtein’s Distance w.r.t to source string.
  - Consider the one which has least Lev. Distance.
  - Result: “Suresh eats mango”
- ▶ Step 4:
  - Compare the frequency of keywords (how many times each occurs) between the source segment and this one.
  - If it is different, Insert the segment in question and its filled target.
  - Else, go to step 5.
  - In the above eg. the occurrences of “Suresh” , “eat” , and “mango” are same in source as well as the segment under consideration.

# TM Building Module.....continued

- ▶ Source segment: “Suresh is eating a mango.”
- ▶ **Step 5:**
  - Repeat the same w.r.t. stop word. (except for “a”, “an”, “the”)
  - If different, Insert.
  - Else, Goto step 6.
  - In above eg. Stop word frequency does not match. So segment will get inserted.
- ▶ **Step 6:**
  - If the frequency of stop words too, matches it would imply that the two segments had exactly same words, with possibly rearranged words.
  - So now the corresponding Hindi translations are checked. If a match is found, segment is not inserted. Only the “freq” is incremented.
  - If not, only the target part of the segment is dumped.

**Thank You**