

**TECHNICAL  
REPORT**

**NEAOSS  
WG3  
TR  
00001**

WG3 N060  
=TR00001:2006

2006-04-13

---

**Information Technology—**

**Requirements for Specification of Input Method  
engine Service Provider Interface**

---

Reference number

NEAOSS/WG3 TR 00001:2006(E)

(empty page)

# Requirements for Specification of Input Method engine Service Provider Interface

## Table of Contents

---

- Foreword
- 1. Scope
- 2. References
- 3. Terms and Definitions
- 4. General requirement
- 5. Functional requirement

## Foreword

---

NEAOSS Forum (Northeast Asia Open Source Software promotion Forum) was formed by China, Korea and Japan governments and regional organizations for OSS promotion; China OSS Promotion Union, Korea OSS Promotion Forum and Japan OSS Promotion Forum. The Forum intends to promote Open Source Software in the northeast Asia area.

NEAOSS Forum formed “WG3:Standardization and Certification Study” in order to study Open-Source Software standardization and certification in Jul. 2004. The Free Standards Group also takes part in the WG3's work as liaison.

NEAOSS Forum WG3 formed subsidiary group:SWG1 in Dec. 2004. It addresses to study Input Method engine Service Provider Interface that can be used worldwide and propose the specification to ISO/IEC JTC 1 through the Free Standards Group, in order to establish a single world wide standard for interface among Input Method engines, User Interface Components and Input Method framework. The establishment of the standard will enable Input Method engines to be bound to Input Method framework on demand.

This document was prepared by the NEAOSS Forum WG3 and is classified as Technical Report type 3.

## 1. Scope

This document addresses the requirements of Interface specifications standard (hereinafter call; IM engine SPI standard) among Input Method Engines, User Interface Components and Input Method Framework. The IM engine SPI standard should realize all general and functional requirements in this document.

## 2. References

**Unicode** ISO/IEC 10646:2003 - Universal Multiple-Octet Coded Character Set (UCS)  
ISO/IEC TR 15285 - An operational model for characters and glyphs  
ISO/IEC JTC1/SC2/WG2  
<http://www.open-std.org/JTC1/SC2/>

Note: ISO/IEC 10646 is also known as the Unicode Standard which is freely available in the Unicode Consortium web site. <http://www.unicode.org/standard/standard.html>

**ISO C99** ISO/IEC 9899:1999 - Programming Language C  
ISO/IEC JTC1/SC22/WG14  
<http://www.open-std.org/JTC1/SC22/WG14/www/standards>

## 3. Terms and Definitions

### 3.0 IM

Abbreviation of Input Method

### 3.1 IM engine

An implementation of one or more input methods logics.

### 3.2 IM logic

A logic implemented in IM engines, such as Chewing or Pinyin methods implemented in a Chinese IM engine.

### 3.3 User Interface Component

An externalized component which handles the user interface related operations on behalf of IM engine, as if it is an extended part of IM engine. User Interface Components and IM engines can be provided and used independently.

### 3.4 IM framework

A framework which manages IM engines and User Interface Components.

### 3.5 IM engine SPI(Service Provider Interface)

An interface among IM engines, User Interface Components and IM framework.

### **3.6 Chewing / Pinyin**

Popular IM logics used in Greater China.

## **4 General requirement**

The IM engine SPI standard should be used for all scripts without modification.

## **5 Functional requirement**

**5.1** IM engine SPI should allow application programs and IM engines to run on separate and heterogeneous platforms.

5.1.1 IM engine SPI should allow conforming IM engines to be neutral to any conforming IM framework.

**5.2** IM engine SPI should provide a mechanism for accommodating IM engine specific extensions.

**5.3** IM engine SPI should be able to accommodate advanced input sources such as handwriting and voice as well as ordinary input sources such as keyboard and pointing device.

5.3.1 Input sources should include any hardware or software input to IM, for example, keyboard press/release event, and candidates list paging/selecting event.

**5.4** IM engine SPI standard should allow subset of the SPI. The way of subset allowed should be clearly defined in the standard.

**5.5** IM engine SPI should allow multiple programming language bindings.

5.5.1 The reference IM engine SPI should be in ISO C99.

5.5.2 Other programming language bindings are out of scope of the IM engine SPI specification.

**5.6** IM engine SPI should be natural language agnostic and should allow the use of multiple languages, multiple IM engines per language, multiple languages per IM engine, multiple IM logics per IM engine and allow a user to dynamically switch among them at runtime.

5.6.1 IM engine SPI should not depend on system's language setting.

**5.7** IM engine SPI should allow IM engines to implement their own licensing schemes.

**5.8** IM engine SPI should allow character data stream/string to be at least in the Unicode coded character set including Variation Selectors.

**5.9** IM engine SPI should allow the control of User Interface Components from IM engine side such as panel, status window and palette.

5.9.1 'Control' includes bi-directional communication.

**5.10** IM engine SPI should provide platform-independent, secure, multi-user handling capability, including data stream management, and storing and retrieving per-user data of reasonable size, such as configuration, dictionary and key bindings.

**5.11** IM engine SPI should allow various input context handling schemes.

**5.12** IM engine SPI should allow IM engine to access and modify data in application domain.

**5.13** The IM engine SPI standard should be specified in multiple parts, a generic interface and architecture specific interfaces.

**5.14** IM engine SPI should specify User Interface Components to be externalized from IM engine and also specify the method to interact between them. IM engine SPI should include the interface between IM framework and User Interface Components which may be in a different SPI from the interface between IM framework and IM engine.

**5.15** IM engine SPI should not preclude non Open Source implementations.

5.15.1 IM engine SPI should include a method to accommodate IM engines' specific key-binding requirements.

*End of document.*