

JMatPro Application programming interface

JMatPro API

JMatPro は、産業界で使用する多成分合金の広範な材料特性の計算を行っています。JMatPro API は、その機能の一部をモジュール化し、それを使用して独自のアプリケーションを開発することができます。

計算モジュール

安定、準安定の相平衡	物理的および熱物性	機械的物性値
凝固の物性値	恒温変態曲線 (TTT)	連続冷却変態曲線 (CCT)
熱処理冷却物性値		

API モジュール

・ Core

一般設定や複数のモジュールに共通する設定に使用される機能を含みます。

・ Solver

逆拡散の有無にかかわらず、多成分合金における安定および準安定相平衡ならびに非平衡 Scheil-Gulliver 計算の熱力学計算を設定および実行する機能を提供します。

・ Coldfire

物理的、熱物理的、および室温のマトリックスの機械的特性の計算を可能にします。

・ Solidification

凝固中の相変化、ならびに物理的および熱物理学的特性の計算を実行する機能を提供します。

・ TTT/CCT

一般鋼の恒温変態曲線(TTT)の計算、連続冷却変態曲線(CCT)の計算を実行する機能を提供します。

・ Cooling

一般鋼の焼入れの、相変態、物理的、熱物理学的、および機械的特性の計算を実行する機能を提供します。

・ Mechanical

室温強度、高温強度、Creep、Flow stress の計算を行います。

開発環境

API は C 言語で書かれており、Microsoft Visual Studio C / C ++ランタイムライブラリがシステムにインストールされている必要があります。 Visual Studio は、Microsoft ダウンロードセンターから入手できます。サポートされているオペレーティングシステムには、Microsoft Windows 7 及び 10 ですが、Linux および Mac OS X はサポートされていません。なお、Python での実行も可能となっています。また、JMatPro 本体のユーザライセンスと共に API ライセンスが必要になります。

日本国内販売元

株式会社ユーイーエス・ソフトウェア・アジア

〒251-0057 神奈川県藤沢市城南 5-5-3 電話: : 0466(34)0562

E-mail imatpro@usi-asia.com Home page <http://www.usi-asia.com/>

アプリケーションのコンパイルと実行

32/64Bit C / C ++開発環境で JMatPro API を使用するアプリケーションを正常にコンパイルするには、コンパイラが必要な Header、Library File が必要になります。これらは、API の Sub directory Include 及び Include file に用意されています。

実行時にアプリケーションは JMatPro Dynamic Link Library(DLL)、設定ファイルと DB、および Sentinel ライセンスキーとそのコントローラファイルにアクセスする必要があります。必要なファイルはすべて、API sub directory Runtime / win32 および/ x64 に用意されています。

正常に実行するには、Sentinel キーが接続されていること、Runtime / win32 または/ x64 を実行可能ファイルと同じレベルに配置しておく必要があります。

凝固計算サンプル Program Solidification / Aluminum

```

// == demo_ss.cpp ==
// Sample code for the JMatPro API Solidification module - Illustrates how to
// set up and run a solidification calculation for a stainless steel
// Copyright (C) 2014-2017, Sente Software Ltd - All rights reserved

```

```
#include <stdio>+
#include <stdlib>+
#include "../Include/jmpcore.h"+
#include "../Include/jmpsolver.h"+
#include "../Include/jmpcoldfire.h"+
#include "../Include/jmpsolidification.h"+
+
int main ()+
{+
    int status; // Optional variable: Holds function return values and allows error-handling.+
    //const int numElements = 11;+
    //char *alloyElements [numElements] = {"Al", "Cu", "Fe", "Mg", "Mn", "Ni", "Pb", "Si", "Sn", "Ti", "Zn"};+
    //double alloyComposition [numElements] = {82.5, 1.0, 1.3, 0.3, 0.3, 0.5, 0.2, 13.0, 0.1, 0.3, 0.5};+
+
    const int numElements = 9;+
    char *alloyElements [numElements] = {"Al", "Cu", "Fe", "Mg", "Mn", "Ni", "Si", "Ti", "Zn"};+
    double alloyComposition [numElements] = {84.12, 1.8, 0.03, 1.8, 0.16, 2.05, 10.0, 0.02, 0.02};+
+
    // Optional function call: Defines the working directory, where intermediate and output files are+
    // written to. Default is "temp/" at the same level as the executable.+
    jmpSetWorkingDirectory ("temp/");+
+
    // Optional function call: Defines whether to print calculation progress to screen. Default is+
    // JMP_FLAG_TRUE.+
    jmpSetScreenOutput (JMP_FLAG_TRUE);+
+
    // Required function call: Sets the material type. Possible choices are defined in jmpcore.h.+
    jmpSetMaterialType (JMP_MATERIAL_ALUMINIUM_ALLOY); +
+
    // Optional function call: Defines whether fast diffusion of interstitial atoms (notably carbon and+
    // nitrogen) should be taken into account. Default is JMP_FLAG_TRUE for steels, JMP_FLAG_FALSE+
    // otherwise.+
}
```

	Temperature (K)	Atomic fraction-LIQUID	Weight fraction-LIQUID	Volume fraction-LIQUID	Atomic
↓	973.15	1	1	1	1
	// Required function	968.15	1	1	1
	jmpSetAlloyElements (	963.15	1	1	1
↓	958.15	1	1	1	1
	// Optional function	953.15	1	1	1
	// JMP_COMPOSITION_UN	948.15	1	1	1
	jmpSetCompositionUnit	943.15	1	1	1
↓	938.15	1	1	1	1
	933.15	1	1	1	1
	928.15	1	1	1	1
	923.15	1	1	1	1
	918.15	1	1	1	1
	913.15	1	1	1	1
	908.15	1	1	1	1
	903.15	1	1	1	1
	898.15	1	1	1	1
	893.15	1	1	1	1
	888.15	1	1	1	1
	883.15	1	1	1	1
	878.15	1	1	1	1
	873.15	1	1	1	1
	868.15	1	1	1	1
	863.15	1	1	1	1
	858.15	1	1	1	1
	853.15	1	1	1	1
	851.82533	1	1	1	1
	848.15	0.955989978	0.956969979	0.9580620355	0.0440080218
	843.15	0.902209467	0.904369497	0.9064803339	0.0977945326
	838.15	0.854467388	0.857687534	0.8603942891	0.145532612
					0.142312466
					0.1396057109

