

# 目录

目录	1
<b>Entegor 产品测试计划</b>	<b>2</b>
1. 简介	2
1.1 目的	2
1.2 背景	2
1.3 范围	2
1.4 项目核实	3
2. 测试需求	4
3. 测试策略	4
3.1 测试类型	4
3.1.1 数据和数据库完整性测试	4
3.1.2 功能测试	4
3.1.3 业务周期测试	5
3.1.4 用户界面测试	5
3.1.5 性能评测	6
3.1.6 强度测试	6
3.1.7 安全性和访问控制性测试	6
3.1.8 安装测试	7
3.2 工具	7
4. 资源	7
4.1 系统资源:	7
4.2 人力资源:	8
5. 项目里程碑	9
6. 可交付的工件	9
7. 附录 A	9

# Entegor 产品测试计划

## 1. 简介

### 1.1 目的

Entegor 的测试计划有助于实现以下目标：

- ※ 确定现有项目的信息和应测试的软件构件
- ※ 列出推荐的测试需求
- ※ 推荐可采用的测试策略，并对测试的工作量进行评估
- ※ 列出测试项目的可交付元素

### 1.2 背景

Entegor 为用户提供企业 IT 系统应用集成的平台。通过这个平台，企业内不同 IT 系统的数据、功能可以以统一的方式，方便地为其他系统共享。系统间由 Entegor 提供的流程机制进行统一调度，实现系统间协作，以完成更高一级的业务功能，达到全面整合企业内部数据和应用系统的目的。

Entegor 由一个中间件服务器，开发环境，iWDL 语言，服务器管理器、客户端应用编程接口和大量通用接口等部件构成。接口负责获得各个 IT 系统功能和数据；服务器运行业务流程，调用接口以驱动外部系统的业务和数据流程，完成不同系统间的信息传递、功能整合；服务器管理器提供对服务器的管理；开发环境帮助用户通过图形化的界面定义业务流程。客户端 API 为客户端程序提供与服务器交流的接口。用户通过客户端 API 与服务器上的流程进行交流。

### 1.3 范围

#### ※ 集成测试

对系统数据接口、功能接口进行集成测试，确保系统个接口之间能够正常通信，保证服务器与用户应用程序界面和其他构件的正常通信，确保连接起来的应用系统的功能和数据按照合理的顺序协同合作，达到业务流程集成目的。

#### ※ 系统性能测试

应用性能测试计划对该系统进行性能测试，确保服务器能够承受最大并发用户操作带来的压力，确保在多 workflow 执行的情况下服务器的响应时间不得低于用户可接受范围，分析性能测试结果，查找系统性能瓶颈，解决问题，优化系统。

#### ※ 系统安装测试

测试产品安装过程的提示语言、错别字和界面情况，确保产品能够安装成功，正常使用。

※ 系统可靠性测试

对系统可靠性进行测试，测试系统平均无故障时间，确定系统最大运行周期。

※ 系统可移植性测试

测试系统移植性，扩展产品对环境的适应力。

**需要测试的功能:**

1. Entegor 工程和业务流程
2. Entegor 服务器
3. Entegor 服务管理器
  - a) 服务器基本信息
  - b) 构造服务器集群
  - c) 服务器管理员信息
  - d) 服务器用户信息
  - e) 工程信息管理
  - f) 流程启动、监视、终止、结果获取等
  - g) 流程日志配置
  - h) 接口管理
4. Entegor 开发环境
5. Entegor 客户端 API
6. Entegor 权限管理

**限制条件:**

本测试计划受限于产品开发人员提交测试的内容和时间的事实。根据开发人员提交模块的实际情况，本计划会做出相应修改。

## 1. 4 项目核实

文档(版本/日期)	已创建或可用	已被接受或已经过复审	作者或来源	备注
需求规约	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
功能性规约	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
用例报告	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
项目计划	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
设计规约	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
原型	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
用户手册	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
业务模型和	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		

业务流程	否	否		
数据模型和数据流	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
业务功能和业务规则	<input type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		
项目或业务风险评估	<input type="checkbox"/> 是 <input type="checkbox"/> 否	<input type="checkbox"/> 是 <input type="checkbox"/> 否		

## 2. 测试需求

见 [《Entegor 测试用例》](#)

## 3. 测试策略

### 3. 1 测试类型

#### 3. 1. 1 数据和数据库完整性测试

测试目标	确保数据库访问和进程正常进行，数据不会遭到损坏
技术	调用各个数据库访问方法和进程，并在其中填充有效和无效的数据（或对数据的请求）。 检查数据库，确保数据已按预期的方式填充，并且所有的数据库事件已正常发生；或者检查所返回的数据，确保为正当的理由检索到了正确的数据。
完成标准	所有的数据库访问和进程都按照设计的方式运行，数据没有遭到损坏。
需考虑的特殊事项	进程应该以手工方式调用 应使用小型或最小的数据库来使所有无法接受的事件具有更大的可视度

#### 3. 1. 2 功能测试

对测试对象的功能测试应侧重于所有可直接追踪到用例或业务功能和业务规则的测试需求。这种测试的目标是核实数据的接受、处理和检索是否正确，以及业务规则的实施是否恰当。此类测试基于黑盒技术，该技术通过图形用户界面 (GUI) 与应用程序进行交互，并对交互的输出或结果进行分析，以此来核实应用程序及其内部进程。以下为各种应用程序列出了推荐使用的测试概要：

测试目标	确保 Entegor 的功能正常，其中包括导航、数据输入、处理和
------	----------------------------------

	检索等功能。
技术	利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用
完成标准	所计划的测试全部执行 所发现的缺陷全部解决
需考虑的特殊事项	产品已完成功能的提交是否在测试范围内

### 3. 1. 3 业务周期测试

业务周期测试模拟一个月内在 Entegor 执行的活动。

测试目标	确保 Entegor 按照所要求的业务模型和时间表正确运行
技术	通过执行以下活动，模拟一个月的业务周期： 将修改或改进对测试对象进行的功能测试，以增加每项功能的执行次数，从而在指定的时间段内模拟若干个不同的用户。  将使用有效的和无效的数据或时间段来执行所有与时间或数据相关的功能。  将在适当的时间执行或启用所有周期性出现的功能。  在测试中还将使用有效的和无效的数据，以核实以下内容： 在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。
完成标准	所计划的测试全部执行 所发现的缺陷全部解决
需考虑的特殊事项	系统时间与服务器时间的不同引起的情况

### 3. 1. 4 用户界面测试

测试目标	Usertasks 用户界面美观，得体，功能正常使用，窗口的对象和特征符合标准。 Server manager 界面美观，得体，功能正常使用，窗口的对象和特征符合标准。 Studio 界面美观，得体，功能正常使用，窗口的对象和特征符合标准。
技术	人工测试或自动化测试工具进行测试，模拟用户操作，确保界面可以正确浏览，并处于正常的对象状态。

完成标准	成功地核实出各个窗口都与基准保持一致，符合可接受标准
需考虑的特殊事项	不同浏览器的测试

### 3. 1. 5 性能评测

性能评测是一种性能测试，它对响应时间、事务处理速率和其他与时间相关的需求进行评测和评估。性能评测的目标是核实性能需求是否都已满足。实施和执行性能评测的目的是将测试对象的性能行为当作条件（例如工作量或硬件配置）的一种函数来进行评测和微调。

测试目标	确认系统服务器所能承受的最大压力 核实系统正常工作量下的响应时间 确定系统性能瓶颈，提优化系统依据
技术	使用 <a href="#">《性能测试用例》</a> 进行测试
完成标准	按照性能测试用例指定，每种情况都成功执行一次。
需考虑的特殊事项	模拟 workflow 数量应与客户环境的 workflow 数保持一致 所使用数据库应与客户环境的数据库一致或相同缩放比例的数据库 考虑网络传输速率

### 3. 1. 6 强度测试

强度测试的目标可表述为确定和记录那些使系统无法继续正常运行的情况或条件。

测试目标	核实系统服务器能够在以下强度条件下正常运行，不会出现任何错误： 服务器上几乎没有或根本没有可用的内存 连接或模拟了最大实际数量的客户机 多个用户同时接管同一个任务
技术	要对有限的资源进行测试，减少或限制服务器上的内存 使用多台客户机运行相同的测试或互补的测试，以产生最繁重的事务量和最差的事务组合
完成标准	测试用例执行完毕，并且在达到或超出指定的系统限制时没有出现任何软件故障，或者导致系统出现故障的条件不在指定的条件范围内
需考虑的特殊事项	注意网络工作强度

### 3. 1. 7 安全性和访问控制性测试

安全性和访问控制性测试侧重于安全性的两个关键方面：  
应用程序级别的安全性：包括对数据和业务功能的访问

系统级别的安全性：包括对系统的登录或远程访问

测试目标	应用程序级别的安全性： 核实 SM 权限管理中定义的各种不同角色的权限 系统级别的安全： 非法用户对该系统的访问限制情况
技术	对应各种权限的角色进行创建，逐个验证。 使用有效和无效用户进行系统访问
完成标准	被规定的角色只得访问权限之内的事务，权限之外的事务不能访问。有效用户正常登录，无效用户给予提示，或警告
需考虑的特殊事项	权限边界处应该特别注意 不拥有该权限的优先级比拥有该权限的优先级高

### 3. 1. 8 安装测试

测试目标	测试 Entegor 可以在 windows 和 linux、unix 操作系统机器上安装并正常使用 测试 Entegor 是否可以成功卸载 检查安装过程中存在的缺陷
技术	手工安装 手工卸载
完成标准	Entegor 产品可以成功安装，产品可以正常使用
需考虑的事项	安装过程中的自定义过程与默认安装的不同测试

### 3. 2 工具

此项目将使用以下工具：

	工具	产商/自产	版本
测试管理	Test Director		
缺陷跟踪	Mantis	自产	
功能测试工具	Quick Test Professional	Mercury	
性能测试工具	Loadrunner	Mercury	
项目管理	SVN		

## 4. 资源

### 4. 1 系统资源：

Server 服务器一台：192.168.0.135

操作系统: English WindowXP  
 CPU: P4 2.4  
 内存: 2G  
 硬盘: 200G  
 系统 agent 服务器一台: 192.168.0.133  
 操作系统: windows 2003  
 CPU:P4 2.4  
 内存: 2G  
 硬盘: 150G  
 Linux 服务器一台:192.168.0.131  
 操作系统: Turbo Linux

## 4. 2 人力资源:

人力资源		
角色	所推荐的最少资源	具体职责或注释
测试经理		进行管理监督 职责: <ul style="list-style-type: none"> <li>• 提供技术指导</li> <li>• 获取适当的资源</li> <li>• 提供管理报告</li> </ul>
测试设计员		确定测试用例、确定测试用例的优先级并实施测试用例 职责: <ul style="list-style-type: none"> <li>• 生成测试计划</li> <li>• 生成测试模型</li> <li>• 评估测试工作的有效性</li> </ul>
测试员		执行测试 职责: <ul style="list-style-type: none"> <li>• 执行测试</li> <li>• 记录结果</li> <li>• 从错误中恢复</li> <li>• 记录变更需求</li> </ul>
测试系统管理员		确保测试环境和资产得到管理和维护 职责: <ul style="list-style-type: none"> <li>• 管理测试系统</li> <li>• 分配和管理角色对测试系统的访问权</li> </ul>
数据库管理员		确保测试数据(数据库)环境和资产得到管理和维护



		职责： •管理测试数据（数据库）
设计员		确定并定义测试类的操作、属性和关联关系 职责： • 确定并定义测试表 • 确定并定义测试包
实施员		实施测试类和测试包，并对他们进行单元测试 职责： • 创建测试模型中实施的测试类和测试包

## 5. 项目里程碑

里程碑任务	工作	开始日期	结束日期
制定测试计划		2006. 12. 20	2006. 12. 23
实施测试	项目测试	2007. 01. 10	
执行测试			
对测试进行评估	测试结果评估		

## 6. 可交付的工件

测试计划  
 测试用例  
 测试 BUG 单  
 测试小结

## 7. 附录 A

以下是一些与测试有关的任务：

制定测试计划

- 确定测试需求
- 评估风险
- 制定测试策略

- 确定测试资源
- 创建时间表
- 生成测试计划

#### 设计测试

- 准备工作量分析文档
- 确定并说明测试用例
- 确定测试过程，并建立测试过程的结构
- 复审和评估测试覆盖

#### 实施测试

- 记录或通过编程创建测试脚本
- 确定设计与实施模型中的测试专用功能
- 建立外部数据集

#### 执行测试

- 执行测试过程
- 评估测试的执行情况
- 恢复暂停的测试
- 核实结果
- 调查意外结果
- 记录缺陷

#### 对测试进行评估

- 评估测试用例覆盖
- 评估代码覆盖
- 分析缺陷
- 确定是否达到了测试完成标准与成功标准