# 第一章 引 言

公安审计是我国审计体系的一个重要组成部分。公安审计机构在各级公安机关的直接领导下，依照国家法律、法规和政策，对本单位的财务收支及其经济效益进行审计监督，独立行使审计职权。审计人员在审计过程起着非常重要的作用，为加强审计人员管理，提高审计人员工作积极性，应对审计人员审计的项目进行考核。正确把握考核的原则、标准、程序和内容，有利于增强考核的效度和信度、提高审计机关的工作效率和审计人员的综合素质。审计机关人员审计项目考核作为审计机关绩效管理的组成部分，在审计机关[人力资源管理](http://www.cnpension.net/rlxc/)活动中发挥着重要的控制、开发、激励和沟通作用。

1.1 公安内部审计现状

长期以来，我国公安内部审计工作是以传统审计为主的，即以内部经济活动单位（部门）为审计对象，以其会计年度所发生的全部经济活动为审计内容，对单位财务收支的真实性、合法性进行审计。传统审计的职能有三个：监督、鉴证、评价，其中监督是最基本的职能，差错防弊是审计最基本的目标。

随着我国财政管理制度的变革以及计算机网络技术在公安机关的普遍应用，内部审计所面临的环境已经发生了很大变化。与其相对应，公安内部审计的审计目标、审计方法、审计手段都与传统审计有很大差异，对内部审计人员素质也提出了更高要求，这也是公安内部审计向更高层次发展的必然趋势

2000年1月起，《中华人民共和国招标投标法》正式实施，这是我国第一部专门规范招投标活动的基本法律，标志着招投标工作步入法制化轨道。2003年1月起正式实施的《中华人民共和国政府采购法》，进一步对政府采购行为作出了法律规范，要求各级国家机关、事业单位和团体组织，使用财政性资金采购依法制定的集中采购目录以内的或者采购限额标准以上的货物、工程和服务，必须通过政府采购形式。这两部法律的实施，对于公安机关财政资金的使用提出了更高、更全面的要求。

一些公安机关实行了会计核算中心制度，取消原有部分会计核算单位，设立报账中心，将原来分属各部门的会计核算、管理、监督等职能委托报账中心统一行使，改变了会计主体以及核算方式。

在实行“收支两条线”后，公安机关执行财经纪律的自觉性已有很大改观，但面临着部门预算、政府采购、招投标等一系列新的要求。如何使公安机关各业务加快适应新的要求？这是公安内部审计面临的一个新任务。当前公安内部审计的工作重点应该从传统的财务收支审计转变到单位管理活动的经济性、资源利用的效率性、达到目的效果性的评价上来。审计目的不仅仅限于差错防弊，更主要是通过评价活动，实现公安机关经济活动的效益最大化。

具体而言，公安内部审计就是要实现“五个转化”：审计职能由监督为主向管理评价转化；审计对象由单位审计向项目审计的转化；审计内容由真实性合法性审计向绩效审计转化；审计手段由手工化向电子化转化；审计时间由事后审计向事中事前转化。围绕“五个转化”，开拓公安内部审计工作新领域:

一是开展专项审计。公安内部审计要以公安业务开展过程中领导和社会关注的热点问题为审计对象开展审计调研，对业务流程的合法、合理、合规性进行评价，提出在业务办理环节可能存在的漏洞，并形成指导性意见，供公安机关领导决策参考。

二是开展绩效审计。绩效审计即前述对于资源利用的经济性、效率性、效果性进行评价。绩效审计的开展，是深化公安机关内部管理的必然要求。它关注的是审计事项的未来，更具有建设性，通过评价，提出促进绩效提高的建议。通过绩效审计，可以促进公安业务创造更好社会效益，也有利于减少人力、物力的挥霍浪费。

三是开展合同审计、招投标审计。这也是内部审计由事后向事前、事中转化的体现。《招标投标法》、《政府采购法》对于采购、建设项目从招投标、签订合同到项目结算都作了详细规定，对具体经办人员提出了更高要求。因此内部审计人员在为各单位提供咨询服务方面应该加大力度，减少在项目实施过程中失误的发生。

在财务信息无纸化环境中，内部审计人员在电脑前，就可以对被审计单位财务情况进行动态适时跟踪，了解被审计单位账面资金流动情况，并进行比较分析，发现疑点及时告知被审计单位，把问题消灭在萌芽状态。同时由于审计手段科技化，也提高了审计工作效率。

公安内部审计新的发展，对公安内部审计人员的素质提出新的要求。审计人员不但需要较高的财务会计知识、了解公安业务，政府决策、治安管理等方面理论知识，掌握计算机网络相关技术，还需要掌握新技术使得贪污腐败的方式越来越隐蔽并多样化的形式。

这些问题的解决，都有赖于内部审计人员自身业务素质的提高。仅仅依靠审计人员自觉提高远远不够，还需要制度的建设，对审计人员进行绩效考核，促使他们掌握更多的知识并应用到公安内部审计当中去。

正确把握考核的原则、标准、程序和内容，有利于增强考核的效度和信度、提高审计机关的工作效率和审计人员的综合素质。审计机关人员审计项目考核作为审计机关绩效管理的组成部分，在审计机关[人力资源管理](http://www.cnpension.net/rlxc/)活动中发挥着重要的控制、开发、激励和沟通作用。。

但受当地经济发展水平、财政管理状况的影响，各地公安机关内部审计发展也不平衡，审计人员的考核方式也不平衡，但是作为今后发展的方向都是一致的。各地内部审计在立足于现在的基础上，认清发展方向，结合当地实际情况，有重点的开展审计及审计考核工作，开创公安内部审计新局面。

1.2 某省公安厅内部审计考核存在的问题

我国财政管理制度的变革以及反腐工作的不断深入，对审计工作提出新的要求，某省公安厅认识到自身审计的不足，加快审计信息化，购买了用友的审计软件进行内部审计。但是审计软件并没有较大地提高审计内部的工作效率，审计项目季度或年度积压率较高。

某省公安厅审计处分析其原因，认识到审计项目的效率依赖内部审计人员自身业务素质，并且审计人员在审计工作中的主观能动性起到重要作用。认识到对审计人员进行考核的重要性。该省公安厅审计处结合借鉴其它兄弟省分公安厅审计处的绩效考核方案，在2008年左右制定了自身的绩效考核方案。

该公安厅的审计采用项目管理的形式进行，其绩效考核方案从自身的特点出发，建立一套基于审计项目的考核方式，分别从基建项目、预算项目，财务收支项目考核、招标文件项目等几种常用的项目形式进行绩效考核设计。每种项目的考核表单都制定不同的考核项和考核规则，如送审金额、审减率、审计时间，发现问题等。

该套实行一年，该公安厅就发现个人通过纸制的申报的绩效的方式流于形式。因为项目会有多人审计，个人申报过程会自动提高自身的分数，多人审计的项目的总分与项目的总分对不上。因为个人单独申报，领导在审批时也很难发现问题，发现了问题，往往会导致审计的多人在总分对总的争论。由于人力的不足，没有专职进行考核的汇总分析统计，到了年终，整个审计处完成多少项目，总送审金额，总审减率都很难统计，更别说统计个人工作的饱和度。

针对于这些问题，该公安厅审计处决定通过采用软件考核系统来解决该问题，考查了市面的大部分绩效考核软件，发现都不能适用，市面的绩效考核软件都是基于个人的考核方式，把考核目标分解成考核指标，通过考核计划进行考核项的汇集。而这种基于KPI的考核方式适用不其基于项目方式的考核。于是其委托软件公司开发了一套了基于项目的考核方式。

经过两年的运行，越来越来发现其现有的系统不能满足日愈发展的审计工作，下面就介绍一下目前系统面临的几个问题：

1．系统扩展性差

审计的业务不够发展，审计的项目类型也在不断地增加，两年来，该公安厅增加了金盾工程项目、预算项目的审计。原系统采用考核项目单独控制的方式。新增项目类型就需要重新开发新功能，开发新功能需要费用需要走发采购招标等相关流程，再加上开发时间，其周期相当漫长。有时没有开发完成就出现新的考核类型，如文宣稿件的考核。

2．不能定制考核表单和规则

该公安厅的考核规则存在逐步完善的过程，每年都会根据系统中的统计情况，结合审计人员自身上报的意给，对考核的打分规则进行微调，使考核规则更公平公正，发挥着重要的控制、开发、激励和沟通作用。原系统的考核表单采用Jsp编写，不能动态调整，其考核规则也是在代码中写死，不能修改。每年都得让开发公司进行表单和规则的定制。过了质保期（应该算新需求），开发公司也不乐意。

3．考核汇总排名不能设定

原的考核排名是简单地把所有考核项目分数相加，按总得分进行排名。但是该考核排名已经满足不了新的需求。现在的排名需要按项目的加权分数进行汇总。并且引入辅助岗位的概念（如该审计处的信息化管理也是某位审计人员兼职完成），不同的辅助岗位在其审计项目的加权比率不同。原系统根本不能实现。

解决以上问题的关键和核心，是应用何种有效的动态控制技术解决系统的动态配置的问题；而BeanShell是基于java的动态语言，动态性是其基础特征。基于BeanShell动态语言的公安厅审计项目考核系统就是解决这些问题最合适的方案。

首先，通过它能真正地实现考核系统模块的解耦，使审计项目类型如插座般地插入和拔出。另外对BeanShell动态语言，考核系统中的考核规则、考核排名不再需要编写编译的代码，而是直接动态编写考核表达式，随时可修改。

因此，将BeanShell动态语言应用于某公安厅的审计项目考核系统中，不但能够实现系统的项目类型扩展，而且还能其它各种不同考核类型的扩展。

1.3 本文的主要内容

本文的目标是将BeanShell的动态语言与公安厅审计项目管理系统有机结合，为某省公安厅考核系统提供强大的扩展能力。使绩效考核更好于调动审计人员的积极性，为加快该省内部审计转型创新，全面实现“职业化、特色化、科学化”发展，充分发挥内部审计服务经济社会又好又快发展的“免疫系统”功能。

本文首先分析了公安厅内部审计的现状和发展，提出我国财政管理制度的变革以及计算机等新技术发展对审计人员的素质要求更高的观点。接下来通过分析某省公安厅审计项目考核的发展现状及其现有系统中存在的问题，确定了基于BeanShell动态语言开发公安厅审计项目考核的定位。充分利用BeanShell的动态性，构建可扩展定制的考核系统。

本文接下来对市面的主流的绩效考核系统进行分析。通过其分析其软件的功能架构设计说明主流的绩效考核系统不适用基于项目类型的考核方式，但同时又可以从中借鉴某些实现方式。把BeanShell引入考核系统，主要使用其动态性，所以本文对BeanShell动态语言的动态性语态进行分析。可以得出BeanShell非常符合当前的项目技术要求。

需求开发是软件系统的必要前提条件，无需求的项目好似“无米之炊”。本文对某省公安厅的审计项目的需求进行了详细分析，分析了其用户基础，分析了其核心流程，最后分析得出其功能模块。需求分析完成之后采用了BeanShell技术对考核系统进行总体的架构设计。从整体框架及数据存储等方面进行总体设计。

基于BeanShell的公安厅审计项目管理系统完全模块化，通过考核类型又可以把所有松散的的模块串在一起，有序地运行。这样就可以把考核数据源、考核表单、考核规则等单独出来。通过注入不同的考核数据源后，再在考核系统中配置其考核表单和考核规则表达式就可以正常运行新的考核项目。

基于BeanShell的公安厅审计项目管理系统，正是为了提高某省公安厅审计系统的扩展性，发挥自身优势，通过构建数据源、表单及考核三大引擎的基础上建立科学的考核模型，实现了基于项目的考核方式，对传统的绩效考核系统进一步发展都有良好的借鉴意义。

1.4 本文的篇章结构

本文共分为五章，首先简要介绍了论文的背景情况，引出了所做的主要工作内容。然后简单介绍了某省公安厅审计项目考核系统的现状和所存在的问题。在此基础上，详细分析了基于BeanShell的公安厅审计项目考核系统的需求，并对系统的实现进行了详细的阐述。最后分析了系统的优点和扩展性。文章的内容具体安排如下：

第一章介绍了国内外的绩效考核系统发展的状况，进而引出了某省公安厅审计项目考核的问题，确定了基于BeanShell的公安厅审计项目考核开发需求，突出了本文的现实意义。

第二章介绍了基于BeanShell的公安厅审计项目考核系统构建的主要技术和方法，主要从基于KPI的绩效考核通用设计和BeanShell的动态语言两个方面来进行阐述。

第三章主要介绍公安厅审计项目考核系统的需求分析。现有绩效考核系统不能满足公安厅的项目考核的业务需求，通过对公安厅审计项目考核系统的用户、核心流程进行分析，提出基于项目的考核设计，并对其功能模块进行说明。

第四章主要对公安厅审计考核系统进行整体的系统结构设计和软件的子系统划分。通过采用BeanShell进行系统解耦设计，降低系统的耦合度。之后再详细地讲解考核数据源、考核规则、考核表单及考核引擎子系统的设计及实现过程。

第五章对全文进行了总结和展望。

# 第二章 考核及BeanShell技术基础

基于BeanShell的公安厅审计项目考核管理系统是针对公安厅审计人员的绩效考核系统，因为审计人员的主要工作是审计项目，所以其绩效考核主要以项目为基础而开展的。而绩效考核系统通过BeanShell技术进行架构设计，可以使用考核的项目类型无限扩展。本章首先通过阐述KPI的绩效考核来了解绩效考核系统的一般设计方案。再进行BeanShell技术的阐述。在后面的章节中会把BeanShell和绩效考核结合起来。

2.1 基于KPI的绩效考核

绩效考核是检查和评定员工对职务所规定的职责的履行程度，以评定其工作业绩。是人力资源管理的核心之一。 绩效考核是人力源管理上不可缺少的工具，它包括直属上级对员工工作的观察和评价。考核的目的并不仅仅是为了奖惩，员工的调任、升迁、加薪等重大决定都必须依据精确的考核结果。

2.1.1绩效考核基本概念

目前世界范围内被广泛谈论和应用的绩效管理的理论方法体系主要有三个，关键绩效指标法（KPI）、是目标管理法（MBO）、是平衡计分卡（BSC）。

关键绩效指标法是将企业的战略目标经过层层分解产生出具体的可操作性的战术目标，通过各指标的达成促成企业目标的达成。所以它主要包含三个方面的内容，即愿景、战略和战术。KPI 将关键的绩效指标与企业的战略挂钩，确定企业为实现其战略而必须实现的指标体系。KPI 是从繁多的绩效指标中提炼出少数关键的指标来进行考核，在减少了对员工的束缚的同时，还大大降低了绩效管理的成本。

目标管理法（MBO）是指组织的最高领导层根据组织面临的形势和社会需要，制定出一定时期内组织所要达到的总目标，然后层层落实，要求下属各部门主管人员以至于每个员工根据上级指定的目标，然后分别制定目标和保证措施形成一个目标体系，并把目标的完成情况作为各部门或个人考评的依据。它的基本思想可以概括为三个方面：以目标为中心、强调系统管理、重视人的因素。但目标管理的缺点是以考核的结果为导向，只注重从结果中分析考核的必要性和缺陷，忽视了对考核过程及方向的监控。

平衡计分卡(BSC)是一个全新的注重企业组织的整体战略实施与完善的管理系统。其最突出的特点是：将企业的愿景、使命和发展战略与企业的绩效评价系统联系起来，它把企业的使命和战略转变为具体的目标和评测指标，以实现战略和绩效的有机结合。

在实际的应用中，常常是将两种方法有机的结合，依据平衡计分卡的方法建立企业KPI 绩效考核体系。依据此方法建立的绩效考核体系，先在美国，然后在整个西方国家得到广泛的应用并取得了巨大成功。

2.1.2 绩效考核的基本设计

绩效考核软件的设计一般包括KPI指标库、绩效计划、 绩效跟踪、绩效方案、绩效分析五大模块，如下图所示：



**图绩效管理框架设计图**

**KPI指标库**

管理人员可设置和维护集团绩效考核业务中的考核指标类型与公共考核指标，支持多级指标类型与多级指标体系，从而构建出集团自身的公共指标库；对暂时不用的考核指标可以封存，需要的时候可以解封指标。可自定义添加并进行任意类型指标分类管理和调用。

**绩效计划**

绩效计划指考核人员或下属考核单位在某个时期内的规划或工作任务，由针对该期间制定的若干绩效目标或工作任务组成。管理人员可制定权限范围内的下属公司的绩效计划、年度考核指标。绩效目标：指某个下属单位在某段期间内要达到或完成的某个业绩效果或工作任务；绩效计划由若干绩效目标组成；对于每个绩效目标，可追溯时间、来源。

**绩效跟踪**

绩效计划下达后，各部门或子公司可以查询自己绩效任务与指标，半年/季度子公司可上报进度与指标，绩效计划发起人可对权限范围内的下属单位的绩效计划的综合进展状况以及计划中各目标的具体进展情况进行记录、查询和浏览。最终汇总年度绩效数据，根据系统设定的打分规则，进行分析评分处理。在绩效执行过程中，可对下属单位的绩效沟通记录和绩效关键事件进行记录，期末时可对整个期间的绩效情况进行总结汇报和查询浏览，具备相关权限人员可进行查询浏览。

**绩效方案**

绩效方案指根据绩效的完成值和指标库而制定不同的角度的考核得分方案，每个考核方案包括方案基本信息、指标通用评分规则、考核结果设置、汇总关系设置、参数设置等信息，存在一次考核中，针对不同业务需要定义不同的考核方案。单次考核中允许设置一个或多个考核方案，如：能力素质考核方案、分子公司领导班子考核方案、中层干部考核方案、日常考核方案等。系统还允许自定义汇总方案，用于对若干基本方案的考核结果进行汇总。

考核量表指绩效计划中对考核对象的考核指标的列表，由针对该对象的若干考核指标及其考评标准组成。通常情况下，每个考核指标在量表中占有一定的考核权重，有些情况下指标没有权重。

**绩效分析**

管理员可对下属单位的各大绩效指标与绩效任务等属性进行自定义查询，并可输出柱状图、报表等分析功能。同时可对下属单位历史考核记录进行汇总分析。可生成对比查询、结果分析、指标分析、绩效综合报表等查询结果。

2.2 BeanShell动态语言

BeanShell是一个小型的，免费的，可嵌入式的，具有面向对象脚本语言特性的Java代码解释器。它是采用Java语言编写，并能够执行标准的Java语句和表达式，还自带简单的脚本命令和语法。它把编程对象当成一个简单的方法，这很像Perl和JavaScript。

BeanShell是可嵌入式的，可以在运行时从你的应用程序调用BeanShell去动态的执行Java代码或是为你的应用程序提供脚本扩展。相反，你也可以从BeanShell调用你的应用程序及其对象，它可以让JAVA对象和API动态运行。

正因为BeanShell是采用JAVA编写的，所以它可以和应用程序运行在相同的JVM空间内，可以自由的传递实时对象的参照(References)到脚本代码中并且作为结果返回。简而言之，BeanShell可以动态的解释JAVA语言。因此它作为你的应用程序的脚本引挚。

BeanShell使用了JAVA的反射机制，可在运行时解释JAVA语句和表达式，可以透明的存取所有的JAVA对象和API。支持JAVA方面和脚本特性。如下表所示：

|  |  |
| --- | --- |
| **JAVA方面的特性** | **脚本特性** |
| 1.覆盖所有的Java语句和表达式语法  2.强效的变量类型与方法  3.原始值到正确数据类型处理能力  4.完全Java算法，逻辑，与按位操作  5.对象构造和多维数组处理  6.具有循环及流程控制结构  7.具有多形/重载特性的方法调用  8.完全的异常(Exception)处理 | 1.动态类型(无类型)的变量  2.具有参数和返回值的脚本方法  3.脚本对象  4.方便的使用JavaBean的属性  5.方便使用原始数据类型的封装器  6.可扩展的工具命令集  7.支持namespace/scope  8.支持this,super,global等关键字。 |

2.2.1动态语言的语法

使用BeanShell，主要编写beanshell的脚本，当然其覆盖所有的Java语句和表达式语法，既然是动态语言，需要更多地了解其动态语言的特性。

**变量的作用范围及可视性**

标准的java程序的变量作用范围是在一个模块中的，而在松散类型的语言中如果在一个模块中没有指定一个变量的类型，则认为是一个全局变量，只要是它以后的代码可以使用该变量，系统在调用该变量的时候自动生成一个全局变量，这就说明了在调用模块之前不能使用该变量。

// Arbitrary code block，Same with any block statement: if, while, try/catch, etc.

//同样也使用于for-loop, if-else等循环语句

{

y = 2; // Untyped variable assigned

int x = 1; // Typed variable assigned

}

print( y ); // 2

print( x ); // Error! x is undefined.

在方法内您可以参考到上下文中上面的变量和方法:

a = 42;

someMethod() { ... }

foo() {

print( a );

someMethod(); // invoke someMethod()

}

// invoke foo()

foo(); // prints 42

如果一个变量只有在方法内使用请定义成局部变量，即加上类型，如果是全局变量请在方法外定义。

var = "global";

foo() {

print(var);// global

String var = "local";

print(var); // local,通过print(super.var);可得到global

}

foo();

print(var);// global

方法内的var（第四行）变量属于局部变量，不会覆盖全局变量var（第一行）的因此改变var（第四行）变量不会影响到全局变量var（第一行）。而使用super关键字可以在局部引用到全局变量。

**装箱和拆箱（box和unbox）**

BeanShell可以采用var指该变量指定为弱类型变量，在运算过程其会自动转为简单类型转换为简单类型。如：

i=5;

iw=new Integer(5);

print( i \* iw ); // 25

但是需要注意的是，对于 2+”3”的表达式，它的计算不是5，而是”23”。对于数字和字符相加，都会转换为字符串进行链接。

**脚本方法**

BeanShell除了可以进行变量的处理，还可以进行方法的处理，可以定义方法象java中的定义方法一样：

int addTwoNumbers( int a, int b ) {return a + b;}

并可以调用它们：sum = addTwoNumbers( 5, 7 );BeanShell变量可以被动态定义为动态类型，方法可以有动态的参数以及返回类型。

add( a, b ) {return a + b;}

在这个方法中，BeanShell将动态的决定类型当这个方法被调用时并且能够准确的计算出你想要的结果：

foo = add(1, 2); print( foo ); // 3

foo = add("Oh", " baby");print( foo ); // Oh baby

在第一个例子中BeanShell将把参数定义为数字型，并返回数字型

在第二个例子中BeanShell将把参数定义为字符型，并返回字符对象

**方便灵活的语法**

BeanShell的动态语法能使用脚本对象，方便的使用JavaBean的属性。可以通过如下三种方式的语法：

|  |  |
| --- | --- |
| **语法方式** | **示例** |
| 标准java语法 | java.awt.Button button = new java.awt.Button();  button.setLabel(“javaButton”); |
| 松散的语法 | button = new java.awt.Button();  button.label = "my button"; |
| 使用{}对一个对象设置属性 | b = new java.awt.Button();  b{"label"} = "my button";  h = new Hashtable();  h{"foo"} = "bar"; // Equivalent to: h.put("foo", "bar"); |

2.2.1BeanShell调用方式

BeanShell采用纯Java编写，可以作为应用程序的脚本引挚。其使用方式相当简单，主要有如下三种方式。第一种是通过BeanShell GUI的方式来调用，该用法使用情况不多。

第二种方式是采用BeanShell命今行/文本的方式，通过java bsh.Interpreter类中的eval方法来运行脚本，如：

Interpreter i = new Interpreter();

i.eval("ss = 4 + 6");

第三种方式是采用BeanShell脚本文件来运行，如：

Interpreter i = new Interpreter();

i.source("script.bsh);

i.eval("ss = add(1,2)");

通过上面的三种方式可以看出，第二种和第三种基本上相同，其只不过多了通过source函数加载脚本文件运行。在实际使用中，一般都是采用第二种方式。