NICE Performance Management

Overview Guide

Version 6.4

Document Revision Date: July 3, 2014

Contents

1 Welcome to NICE Performance Management

This document provides an overview of the NICE Performance Management (NICE PM) functionality and architecture, system requirements, and how to get started using the application. It is written for anyone interested in NICE PM.

This chapter explains how NICE PM adds value to your organization by integrating with your data sources, transforming and consolidating the data, and presenting the data in dynamic reports and dashboards that give users visibility into employee, group, and operational performance. Armed with new insights about key performance measures, users can leverage NICE PM workflows to drive performance improvement throughout the organization.

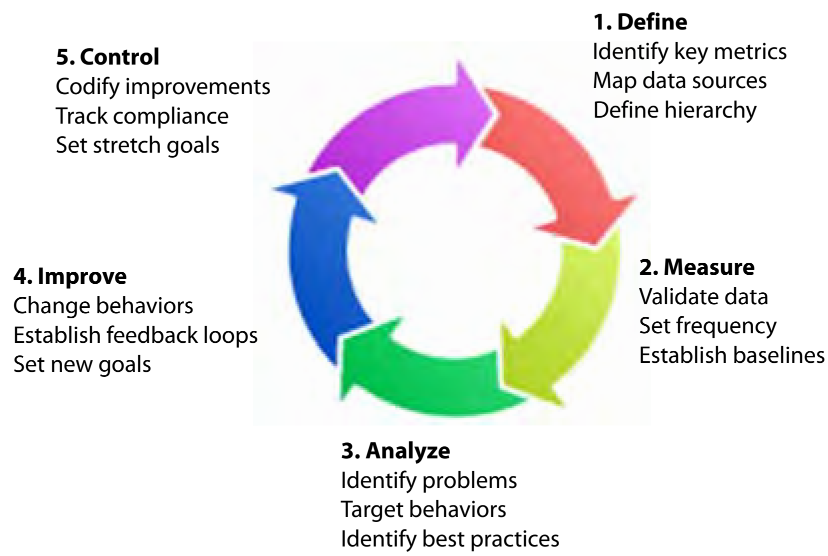
About NICE Performance Management

NICE PM is an enterprise application that allows users to view metrics about their performance and initiate targeted coaching sessions based on that data. It is designed to turn customer operations into high-performing, data-driven organizations. NICE PM increases operational efficiency and effectiveness by consolidating data from disparate sources, providing advanced analytics, delivering personalized dashboards to end users, and integrating workflow and process improvement tools for every role in the organization. All this functionality is delivered to users right in their web browsers without their having to install any software, making rollout across your organization fast and easy. You can host NICE PM on application servers behind your firewall or use the version hosted by NICE Systems.

Instituting cyclical, ongoing improvement

Data analysis, dynamic reporting, and NICE PM-driven workflows all combine to enable an ongoing cycle of self-improvement, as shown in the following graphic. The cycle includes defining data sources and metrics, establishing performance measurement, analyzing the results of that measurement, addressing any problems, and extending the improvements throughout the organization and tracking compliance.

The result is an organization committed to constant self-improvement and enjoying the benefits that the process brings.



NICE PM enables customer organizations to:

- Facilitate self-improvement in agent performance.
- Track performance over time with account organization changes such as additions, subtractions, and transfers over time.
- Improve agent coaching.
- Provide management tools to analyze performance issues.
- Provide standardized best practices for coaching, recognition, hiring, and automating across the organization.
- Automate reporting and analysis processes.
- Automate incentive plan calculations.
- Provide a data foundation for Six Sigma, COPC, and other process improvement initiatives.

Collecting and transforming data

NICE PM aggregates data from a wide range of customer data sources and consolidates it into a data mart that serves as the single, trusted source of truth about operational and employee performance. NICE PM integrates with existing systems such as automatic call distribution systems (ACD), work force management systems (WFM), customer relationship management systems (CRM), human resource systems, and others. NICE PM can obtain data through spreadsheets and flat files or by connecting directly to databases or other data warehouses with ODBC connections.

Data is also entered directly into the system by users through forms used in the context of everyday user and administrative tasks. Data entered through NICE PM forms can be analyzed side-by-side with integrated source data.

Measuring performance

NICE PM uses the aggregated data as the basis for measurements of employee and operational performance. Measurements are defined in metrics using a powerful language that supports a wide range of goal, key performance indicator, and incentive calculations.

Examples of NICE PM functions that support performance measurement include:

- Status metrics that display graphical icons, such as traffic lights, thumbs (up and down), and check marks, as a result of comparisons against a value.

	Metrics	
Agents on Team Reggie Putnam	AHT	AD_Status
Alphonse Dove	566 sec.	✓
Anton Mckenna	606 sec.	✗
Ashlee Lyles	1,201 sec.	✗
Brandy Benavides	324 sec.	✓
Cora Bowen	1,000 sec.	✗
Danette Dawson	759 sec.	✗
Dena Proctor	562 sec.	✓

- Business functions such as BalancedScore, which computes a weighted average from up to six inputs, PctMetGoal, which shows the percentage of people who met a specified goal for a specified time range, and Scale, which is used to scale dissimilar measures for comparison.

Analyzing and presenting data

Goals, KPIs, and incentive metrics are presented to users on a personalized basis through dashboards and reports specific to the user's role. For example, agents and supervisors view dashboards that present only the most important measures. Managers and analysts who want to ask deeper questions about performance metrics have access to powerful analytic reports that enable them to view the data in different ways: by adjusting and swapping the dimensional axes of tables and charts, drilling down into the data for detailed analysis, or rolling up to a bird's-eye view.

Following are typical examples of how users manipulate a report to answer questions:

- Examine performance at the team level, and then drill down to see details for specific agents.
- Adjust the time axis to view by month, by quarter, by year.
- Change the location dimension to view the performance of all agents by site, by region, by country.
- See something interesting in the performance of one region and drill down for a more detailed look.
- Adjust the settings so the report accounts for personnel changes over time.

Driving business workflows

Organizations see the full value of NICE PM when their employees, supervisors, and managers—given more insight about their individual and team performance—implement performance-improvement strategies. Automated workflows help standardize management practices across teams, locations, and operations. Workflows facilitate employee and performance management activities, such as employee coaching, recognition, probation, coming on-board, and quality audits. They allow supervisors to focus on critical employee development opportunities and reduce administrative tasks.

Workflows can be initiated by system alerts that occur when key performance metrics cross defined thresholds, or users can initiate workflows as part of their normal activities. NICE PM-driven workflows are valuable to staff in multiple levels within the organization because they drive performance efficiencies by ensuring that tasks are completed in a reasonable time frame.

Administering and maintaining the system

NICE PM delivers a comprehensive set of development and administration tools that enables business users to maintain, customize, and extend the application. Authoring tools available through a browser enable customers to define metrics, set up alerts and messages, design reports, dashboards, and forms, publish content to the portal, and manage user permissions, in addition to many other administrative tasks. A range of data management tools makes it easy to maintain organization chart hierarchies for reporting, role-based personalization, security, and workflow while still supporting maximum customization and flexibility.

Supporting large, complex enterprise deployments

NICE PM is a browser-based, n-tier application designed to support tens of thousands of users. With its simple administration tools, the application is quick to deploy and easy to maintain. Developed in Java, the application's open-standards-based architecture meets the demands of even the largest enterprises, ensuring the lowest total cost of ownership. No special client software is required beyond a standard browser (Internet Explorer, Firefox, or Safari). For the back-end databases, NICE PM supports Oracle and Microsoft SQL Server.

2 Architectural Overview

This chapter provides an overview of the NICE PM architecture. In this chapter, you'll learn how NICE PM is designed to help your company meet its performance management needs.

This chapter contains the following sections:

About the NICE PM architecture

At the highest level, NICE PM is made up of two key components:

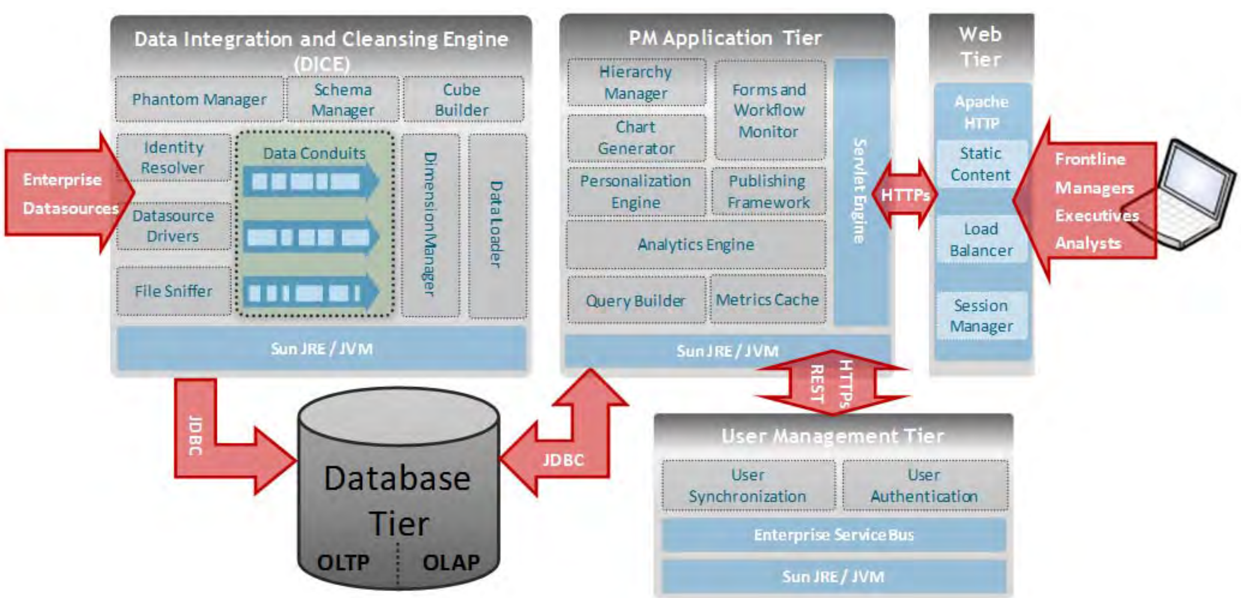
- The **application**. This includes all the performance management functionality, such as coaching and observation forms that users fill in to provide data, reports and dashboards that display key performance indicators, and administration screens for managing users and the application. It comes with pre-built content that can be used out-of-the-box, including data connectors and reports. You can easily publish additional content through the Publishing Framework. It also includes the data mart, which aggregates data from external sources and forms within the application.
- The **platform**. This includes the core infrastructure for processing and displaying the functionality available in the application. The platform includes the browser-based portal that displays content personalized for users by their role in the organization. The service platform is comprised of a powerful and flexible enterprise service bus (ESB) that carries data between

services, such as user management and authentication, and the rest of the application. The platform also includes a workflow framework, which provides the ability to create powerful workflows for routing alerts, messages, and tasks, and an intuitive user interface for building forms that can be integrated with the workflow.

At a more detailed level, the logical architecture is made up of the following tiers:

- Web tier: the web server, load balancer, and session manager, which act as the interface between the end user (client) and the application and pass requests between them.
- Application tier: all the performance management functionality that users see in the portal, as well as the underlying engines that process and serve up the content.
- User management tier: the User Management service handles user synchronization and authorization among applications. It uses an enterprise service bus to handle the messaging associated with managing users, such as sending password notifications to users.
- Database tier: the analytical (OLAP) and transactional (OLTP) data marts, which store and provide data for reports and searches.
- Data integration tier: the data integration and cleansing engine (DICE) handles all the incoming data from external sources and then populates the data marts.

Figure 1: Logical architecture



The ESB is the master application and all others, including the Liferay portal, are slaves. Specifically, Liferay is a passive slave application because new users can be pushed to it from the user management area in the ESB. The other slave application is NICE PM, but whether it takes the passive or active role depends on the configuration.

For example, in a NICE PM deployment, both NICE PM and Liferay are passive. If a user is created in a subordinate application, the record is synchronized to the ESB and pushed automatically to the passive applications. If a user is updated in a subordinate application, the record is synchronized to the ESB and to all slave applications in which the user is defined, including the portal.

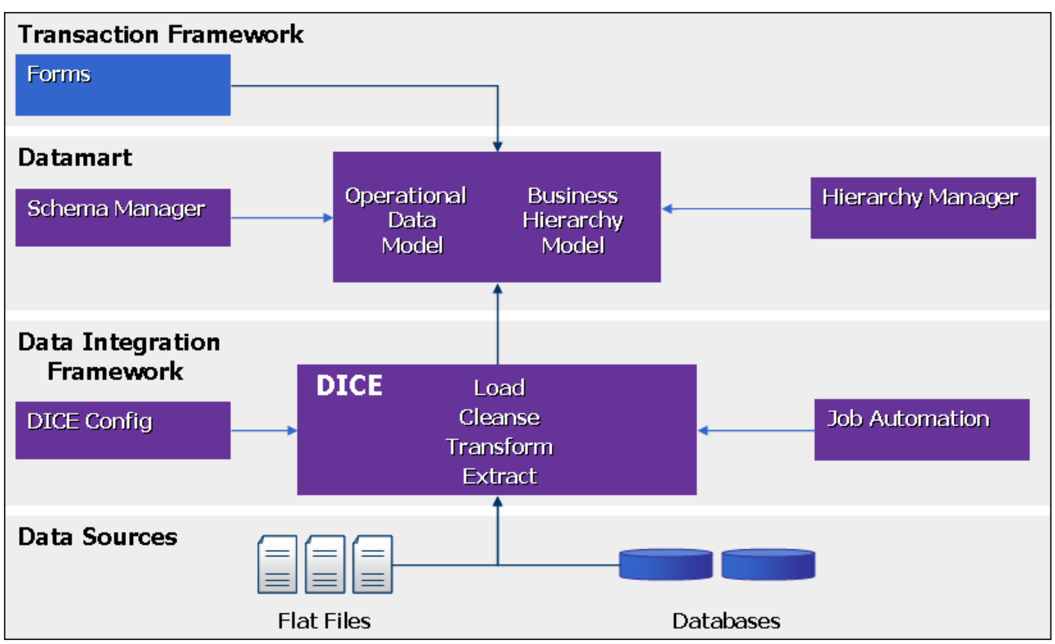
In a NICE ICM deployment, both NICE PM and Liferay are passive. If a user is created in a subordinate application, the record is synchronized to the ESB and pushed automatically to the passive applications. If a user is updated in a subordinate application, the record is synchronized to the ESB and to all slave applications in which the user is defined, including the portal. NICE ICM is never passive.

The rest of this chapter describes the frameworks and components in the architecture in more detail.

Data mart and Data Integration Framework

Data from source systems and data collected through forms is stored in the data mart. Data is stored in relational tables in a two-part data model on which Analytic, Workflow, and Transaction Frameworks operate. The two parts of the data model are the Operational Data Model and the Business Hierarchy Model.

Figure 2: Data mart and Data Integration Framework



The Operational Data Model contains data about events or transactions that is either integrated from external systems or generated through forms. For example, the Operational Data Model stores measurement data from ACD systems or coaching data from coaching forms.

The Business Hierarchy Model contains data that describes the organizational structure and entities about which measurements are made. For example, the Business Hierarchy Model might describe which department an employee is in, or which region a customer is part of. The Business Hierarchy Model is a key component of many aspects of NICE PM applications. It is the foundation for OLAP dimensions and determines how metrics can be sliced and diced in OLAP reports. The Business Hierarchy Model is also used for alert routing in the Workflow Framework and personalization and security in the Publishing Framework.

The platform-based architecture is a powerful and flexible architecture that allows you to build and configure database tables to develop the schema without having to go into the Relational Database Management System (RDBMS) itself. Data mart tables are defined using metadata objects called *metabean types* that specify column names, data types, and foreign key references. A process called `schemaload` takes each registered metabean type and creates a database table in the format required by the RDBMS.

The Hierarchy Manager is a specialized tool for importing and exporting data between tables represented in the Business Hierarchy Model and MS Excel spreadsheets. This tool provides a simple user interface for configuring import and export parameters, generating an XLS template for storing the data, and running the import and export routines. Using this tool, a user can easily seed their hierarchies with data from external systems and modify large or small volumes of hierarchy data using the closed-loop export-import routine.

The Hierarchy Manager also provides wizard-based tools for creating or modifying information specific to person hierarchies (sometimes referred to as Org Chart), including adding a new employee or group, changing an employee's manager, terminating an employee, and managing an employee's identities in the system. For more information, see "About the Person dimension" on page 24.

Data Integration Framework

The Data Integration and Cleansing Engine (DICE) is the core of the Data Integration Framework. DICE captures data from source data systems and integrates it into the data mart. The DICE conversion process uses the ETL model to transform data from diverse sources into a common format usable by other parts of the NICE PM system. The DICE engine is specially designed to handle two common challenges of integrating performance data. First, it manages identity assignments such that entities (typically people) from different source systems (for example, HR and WFM systems) can be matched and reference by a single NICE PM identity in the data mart. Second, DICE can import data about entities that haven't been set up in the system yet, storing that "phantom data" in a temporary structure until it can be properly assigned.

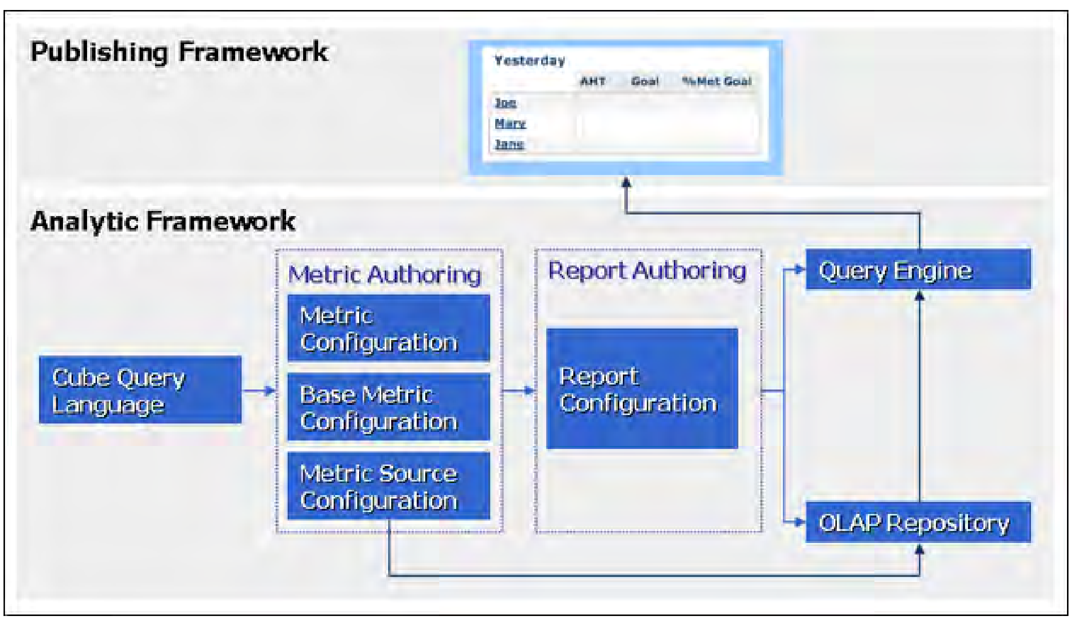
A DICE job refers to a specific conversion configuration that specifies a data source, target table in the data mart, and a mapping between the two. Customers can configure and schedule multiple DICE jobs, depending on their specific needs. Additionally, DICE jobs can be scheduled for different frequencies. For example, some data sources, such as a spreadsheet, might be a one-time conversion during initial product implementation. Others, such as a sales database, might require weekly or even daily jobs, depending on how rapidly the data is changing.

In addition to the data integration framework, you can capture data that users enter into forms in the application and write that data to the data mart. Users can also retrieve information from the data more through transactional reports and searches. For more information, see "Forms" on page 19 and "Transactional reports and searches" on page 20.

Analytic Framework

The Analytic Framework is responsible for generating the OLAP reports that are used for presentation, analysis, and monitoring of performance. Reports present detailed views of performance metrics along specified business dimensions and enable users to view the data in different ways by drilling down and rolling up along the axes of the report.

Figure 3: Analytic Framework



Reports

Reports can be powerful and dynamic analytic tools, or they can be presented in a more static fashion as components of integrated performance dashboards. Reports are authored by administrators or analysts using an intuitive graphical user interface (GUI). Views of a given report can be personalized by role so that users of the report see measurements only about the people, groups, or other business entities they are interested in.

My Teams	Jan-2005	Feb-2005	Mar-2005	Apr-2005	May-2005	Jun-2005
Team Ahmed Tang	48	44	48	15	0	0
Team Arline Whipple	52	62	56	12	0	0
Team Athena Stoddard	51	54	53	14	0	0
Team Carolina Schwartz	75	81	66	6	0	0
Team Gonzalo Olsen	60	70	70	21	0	0
Team Jeff Metz	63	54	67	18	0	0
Team Joann Brand	84	84	77	27	0	0
Team Lou Jamison	63	45	54	12	0	0
Team Mandy Durbin	43	46	41	11	0	0
Team Meghan Rushing	0	0	0	17	0	0
Team Nathanial Huff	58	53	57	11	0	0
Team Octavio Friend	37	46	41	11	0	0
Team Paul Carmen	0	0	0	0	0	0
Team Reggie Putnam	92	106	101	34	0	0
Team Rosalee Pryor	52	46	43	7	0	0
Team Stan Varela	50	51	57	20	0	0
Site Evals	828	842	831	236	0	0

Key features in reports that enable insightful analysis include:

Metrics. Metrics are the performance measurements that appear on a report. They provide measurements for people, groups, or other business entities represented in dimensions. Metrics can be simple representations or aggregations of values stored in the data mart, or they can be complex calculations that entail sophisticated multi-level and multi-dimensional operations.

Dimensions and axes. Dimensions represent the business entities being measured and appear as axes in the report. Users can swap axes and drill down or roll up on axes to different levels in dimensional hierarchies, allowing maximum flexibility for shaping the data in the report.

Yardsticks. Yardsticks provide context for the report data by revealing comparative information. For example, yardsticks can show minimum values, maximum values, standard deviations, or averages relevant to metrics on the report.

Filters. Users can create filters that limit the data that appears in the report, such as excluding employees in certain roles or excluding calls during certain periods.

Phrases. Phrases are pre-configured constraints such as Last Week or People Below Me that adjust the information displayed on the dashboard.As such, a phrase determines which entities from a dimension should appear as the labels for the x-axis (columns), y-axis (rows), or z-axes (corners) of a report.

Phrases can be absolute or relative. Absolute phrases define fixed points in a dimension, such as March 30 or Agent Joe Smith. Relative phrases, which are only possible in the Time and Person dimensions, define points relative to the variables "Today" and "The Logged in User," respectively. Relative reports drastically reduce the number of reports required to send personalized results to thousands of different users over time.

Drill paths. Drill paths define the levels of a dimension a user can drill down to and roll up from the saved (default) view of the report.

Charts. A report can depict data both graphically through charts and numerically through tables. The same report can include both table and chart formats. Available chart formats include bar, line, pie, dial, and many others.

OLAP engine

OLAP reports are generated by the OLAP engine, which has several key components:

OLAP repository

The OLAP repository stores the OLAP data used in reports. The OLAP data mart consists of *dimensions*, which are hierarchies such as Person and Time, *facts* or *cubes*, which store data from chunks as defined by base metrics aggregated across multiple axes, and *spaces*, which store calculated metrics. Each axis draws its values from a specified dimension, so a cube depends on the dimensions that its axes require.

Base metrics and metric sources

The fact tables in the OLAP Repository are based on base metric configurations. Each base metric is based in turn on a specific metric source. A metric source describes a table in the data mart, defining the columns that hold dimension data (which become axes on a report) and fact data (which become base metrics). Metric sources make it easier to create base metrics since they eliminate the need to enter axis information for every base metric that comes from a given table. They also provide an abstraction layer that hides extraneous details in the underlying tables from the base metric configurer.

Base metrics are the building blocks of all metric calculations. A base metric configuration pulls a single fact source from a metric source and applies additional settings, such as the units, the default aggregation method (such as sum, min, or avg), and whether drilling through to transactional data is allowed from that metric.

CubePop

The OLAP Repository is populated and updated by a process called `cubepop`, which reloads both the fact tables and the dimension tables. If dimension definitions or data have not changed, customers can use `nextpop`, which reloads only the fact tables.

Calculated metrics and the Cube Query Language

Calculated metrics generate the analytical results that appear in dashboards and reports. You define calculated metrics by writing expressions in Cube Query Language (CQL) to combine base metrics and other calculated metrics. CQL is a powerful and flexible expression language that is capable of performing all kinds of calculations on data drawn from the OLAP repository. The language supports three classes of functions:

Basic: These functions include standard arithmetic, statistical, and logical operators.

Advanced: These functions perform complex operations. For example, some advanced functions allow you to change the cube's shape; that is, to change the cube's frame to include or exclude an axis or to alter the grains along an axis. Other advanced functions allow you to compute ranks and compare values over time. Advanced calculations are often required for determining goals, incentives, utilization rates, and other metrics that require data aggregations from different levels in a dimensional hierarchy to be calculated.

Business: These functions can execute sophisticated expressions with multiple advanced operators, but they abstract this complexity from the configurer, requiring only a few simple arguments. For example, NICE PM provides higher order functions such as rank, BalancedScore, and PctMetGoal.

Workflow Framework

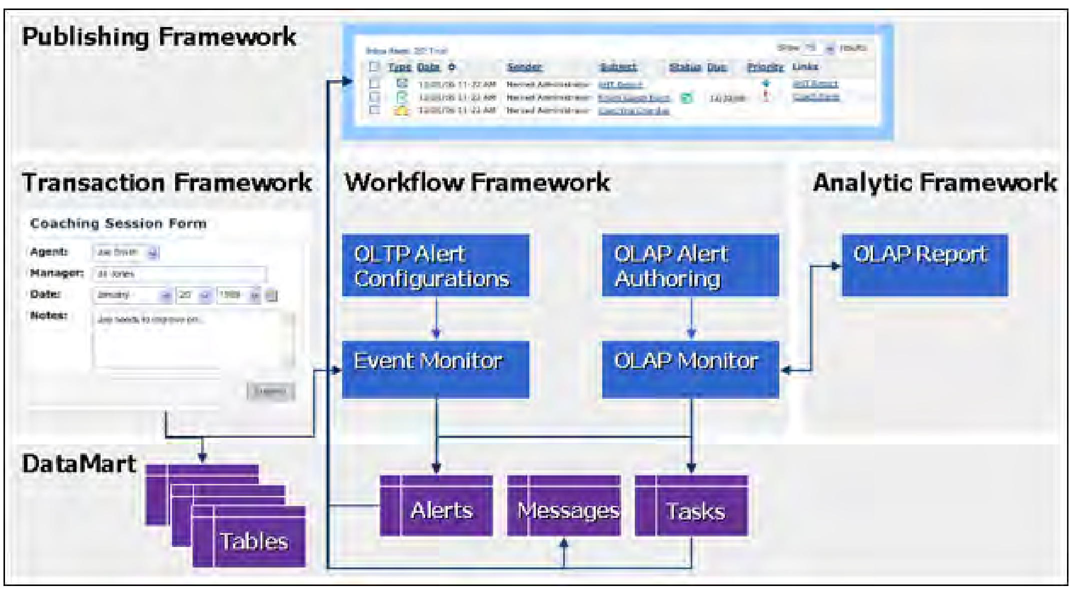
A workflow (also called a *process*) is used to define how data flows through the application, including which forms will appear, how to capture data along the way, and how the data is routed. NICE PM workflow components provide users the ability to take action based on insights derived from performance measurements. The primary components of the Workflow Framework are called Alerts, Messages, and Tasks (AMT).

Alerts are system-generated notifications that warn of extraordinary conditions in the operation, such as when an employee is failing to meet goals, or events such as when a supervisor submits an evaluation form. There are two types of alerts: OLAP alerts and data mart alerts. OLAP alerts are triggered by a monitoring engine when OLAP metrics meet specified conditions, such as `AHT > Goal_Threshold for Agent Joe Smith`. Data mart alerts are triggered by a different monitoring engine when updates in the data mart meet specified conditions, such as when a form record is submitted.

Messages are notifications sent by users to other users of the system. Messages can be sent with links to reports or dashboards and are typically used for informal communications or announcements to team members. For example, managers might send a message to their team members to tell them about a group function.

Tasks are specific types of alerts or messages that require an employee to complete some action, such as filling out and submitting a form. Tasks contain a due date and a status that indicates whether they are complete or not.

Figure 4: Workflow Framework



By coupling AMT with Transaction Framework components, such as forms and transactional reports, NICE PM is able to provide closed-loop support for guided and ad hoc user workflows. AMT inbox components can be published in various places in the end-user portal, ensuring that critical workflow information is always visible to end users. An alert authoring tool is provided for configuring OLAP-based alerts.

Examples of workflows include the following:

- **Customer satisfaction:** A customer satisfaction workflow might initiate and track follow-up calls. A message schedules the action and an alert is sent to remind the agent when the follow-up call is due.
- **Internal human resources:** An internal workflow might process and manage vacation requests. An agent creates a request with a form, an action which in turn triggers a task for the supervisor: reviewing and approving the request. This task is scheduled to be completed within a time period, and the supervisor is reminded via an alert when that time period expires.
- **Employee coaching:** A coaching workflow might be initiated when a specific key performance indicator falls below a certain point. This event requires the supervisor to schedule a coaching session to address the issue within a set time period. Notes from the coaching session are subsequently entered into NICE PM using a form. This action triggers a follow-up session to check any improvement.

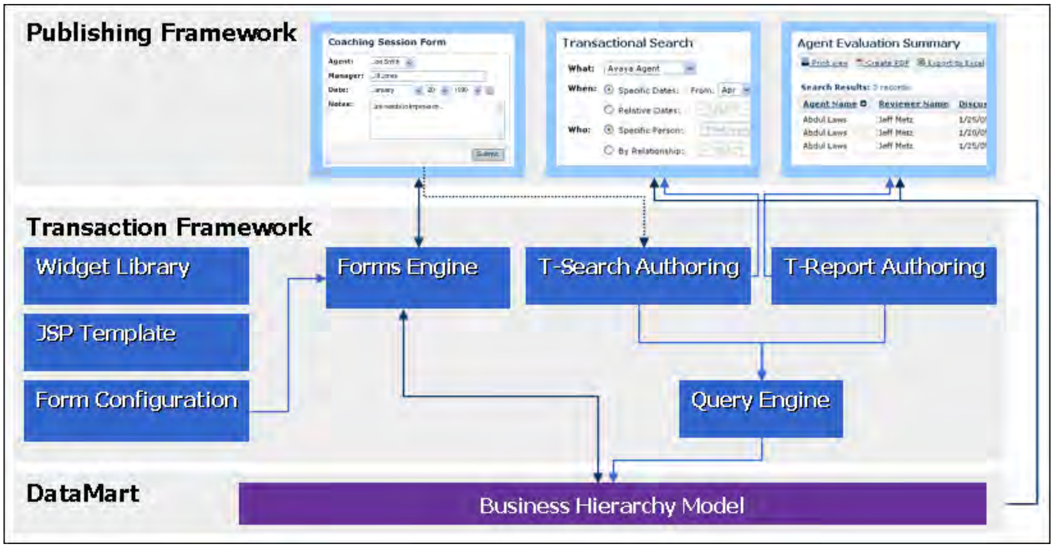
Transaction Framework

The NICE PM Transaction Framework is a complete system for creating, reading, and updating data in the NICE PM data mart, most typically data from transactional tables. The transactional system supports user activities such as:

- Completing a performance review
- Recording a disciplinary action
- Recording a coaching session
- Creating and tracking goals
- Tracking time

The three primary components of the Transaction Framework are forms, transactional reports (T-reports), and transactional searches (T-searches).

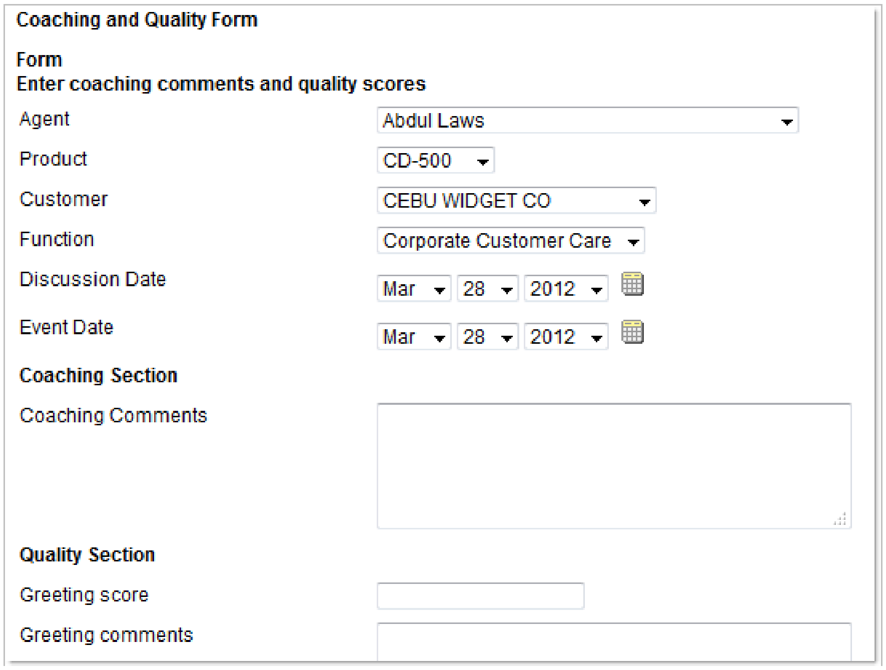
Figure 5: Transaction Framework



Forms

Forms provide an easy and structured way for users to enter information. For example, if you want to make sure that your employees create status reports in a consistent format, you can create a form that allows them to select predefined project codes from a list, automatically enters their name on the report, and so on. You can even set up an automated alert to remind them to fill in their status report each week.

Forms enable users to enter data into or retrieve data from the OLTP data mart. They support both data input and read-only data display. Input forms support user-interface-level data validation, and custom business logic can be attached on form submission events through workflows.



The visual form-authoring tool allows you to quickly and easily create forms by dragging and dropping components such as lists and check boxes to a canvas.

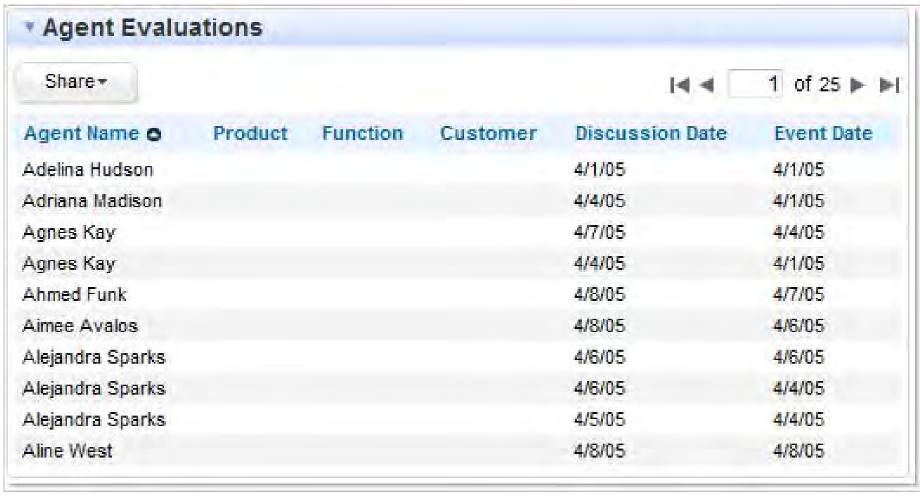
Transactional reports and searches

You can use reports to obtain views of data that is aggregated and calculated into data cubes. This is called analytical or OLAP data. You can use transactional reports and searches, on the other hand, to search for and view transactional or OLTP data. Transactional data is extracted from forms or external databases and processed by DICE.

You can configure and make available both transactional reports, which retrieve data that matches pre-specified criteria, and transactional searches, which retrieve data that matches criteria the user specifies.

Specific transactional search topics and search criteria are set up during system configuration. Typically, a search topic specifies a specific data source (such as a form or an extracted data set), together with relevant search criteria.

When you create a transactional report or search, in addition to the system default criteria, you can specify custom criteria that define a set of records to retrieve from the data tables. For example, you can create a custom form and search for the data entered in that form.

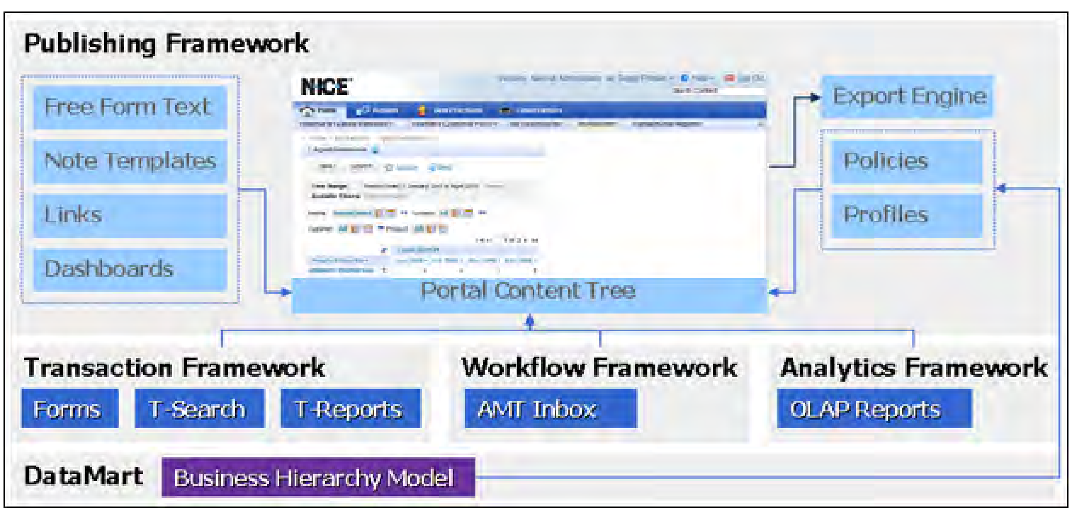


Agent Name	Product	Function	Customer	Discussion Date	Event Date
Adelina Hudson				4/1/05	4/1/05
Adriana Madison				4/4/05	4/1/05
Agnes Kay				4/7/05	4/4/05
Agnes Kay				4/4/05	4/1/05
Ahmed Funk				4/8/05	4/7/05
Aimee Avalos				4/8/05	4/6/05
Alejandra Sparks				4/6/05	4/6/05
Alejandra Sparks				4/6/05	4/4/05
Alejandra Sparks				4/5/05	4/4/05
Aline West				4/8/05	4/8/05

Publishing Framework

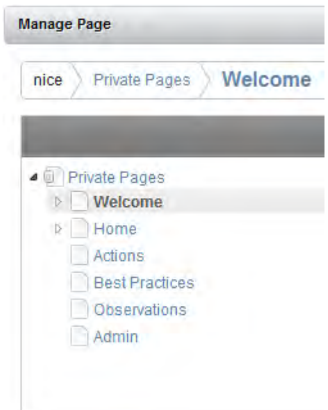
The Publishing Framework controls the end-user experience. Within the Publishing Framework, content items such as reports, dashboards, transactional searches, and forms are published to end users on pages created in the application portal. Access to and personalization of these items are controlled by profile groups, which categorize users based on roles. Alerts, messages, and tasks are exchanged using AMT inbox items, and integration with the offline world is supported through the Print, PDF, and Export capabilities.

Figure 6: Publishing Framework

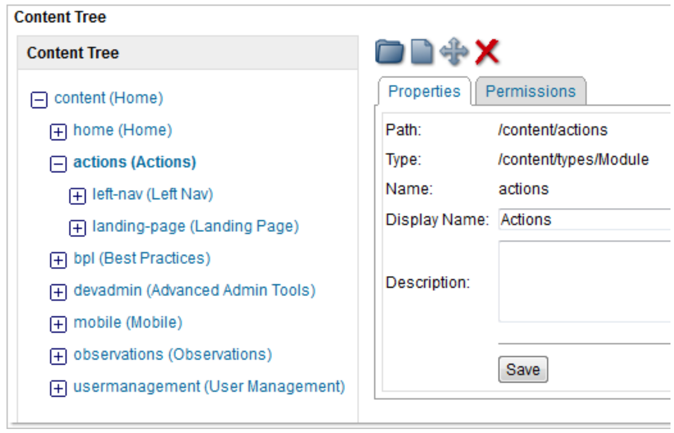


Application portal

The application portal is the interface that displays all application components to the end user. The portal is structured hierarchically, with pages or folders at the top level, each of which contains landing pages as sub-levels, and any number of pages beneath. You can create pages for publishing content items on portlets in the Liferay portal (or the content tree for the mobile application) using tools provided in the administrative user interface.



Liferay



Content tree (for mobile content)

Personalization and security

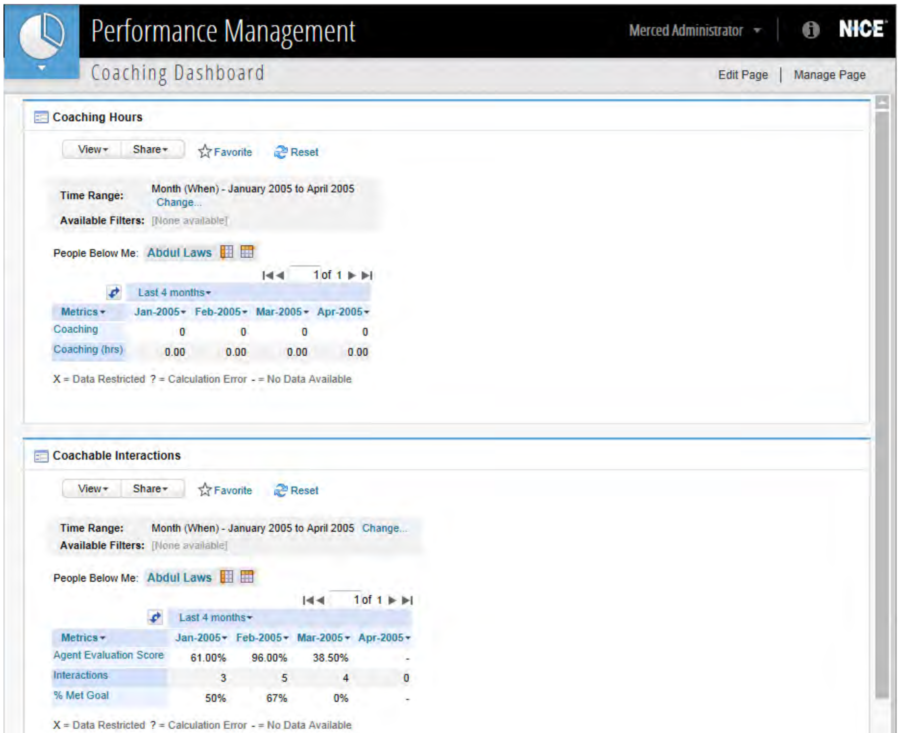
Administrators define profiles that group together people with similar attributes, such as those with the same job title, department, or skills. These profiles are then used to personalize and control access to the portal by setting permissions on the different content items published on the portal. For example, you can publish a report for executives, but not for agents, to the home page. You can hide the report from agents by setting the security and view policies that limit access to the report only to users with the executive profile.

Profiles are defined based on criteria contained in the Person dimension, so there is no need for users to be manually added to, deleted from, or moved within groups. This happens automatically when org chart updates are made.

Dashboards

Dashboards are a specific portlet type that combine related reports and other content items onto a single page, enabling users to access all the information and tools they need for a topic in one convenient place. Dashboards show a static snapshot of the data, so the display isn't cluttered with controls for navigating within reports. To analyze more deeply or to work with information presented in a content item, links are provided to the individual items.

Dashboards are configured in a powerful dashboard authoring tool that enables content administrators to adjust the layout and visual properties of different content items, optimizing the dashboard display for different uses. For example, it is possible to modify the sizes of charts, hide portions of a report, and choose whether to display content in rows or columns in the dashboard.



Using print, PDF, and export options

Users can archive or share data from reports, dashboards, transactional reports, and forms using the following options:

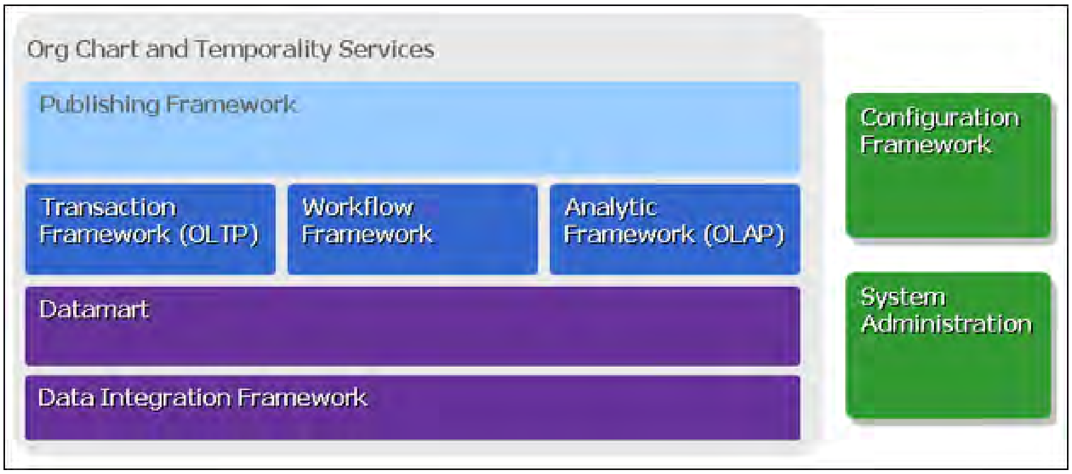
- **Print:** Users can print content items by opening a window that shows a view formatted for printing and allows users to insert custom formatting.
- **Create PDF:** Users can specify custom formatting and create a PDF file of the presented report. Users can view, print, and save the PDF file.
- **Export to Excel:** Users can export the data in a report to a Microsoft Excel spreadsheet, either a formatted file useful for viewing, or an unformatted file useful for importing into other applications.

About the Person dimension

The Person dimension, also referred to as *Org Chart*, is central to the functioning of the application. All the parts of NICE PM depend on org chart data and services that draw upon that data.

The Person dimension is based on your org chart, which contains information about all the employees in your organization and their relationships to one another. The org chart shows the employee hierarchy, and NICE PM uses this information in many ways.

Figure 7: Org Chart Services



NICE PM's collected and stored org chart data enables you to:

- **Create users.** Each person in the org chart can have a user name and password. Therefore, everyone about whom the system collects data can potentially access the system and view data.
- **Limit functional access.** Each user is assigned a role based on the org chart. A user's role determines whether that user can access a report, dashboard, directory, or form. For example, users can only complete certain forms about employees on their team.
- **Limit data access.** Users can be restricted to view data only about people related to them in a certain way. For example, supervisors can be configured to view data only for the agents they supervise.
- **Personalize reports.** Many reports show data relevant to users and their role. For example, supervisors viewing a report see data about their team, whereas directors viewing the same report might see data about their center.
- **Route alerts, messages, and tasks.** Most workflows use the employee hierarchy, sending alerts, messages, and tasks to an employee's boss, and then to that boss's boss.

Tracking data changes over time

NICE PM is unique among OLAP-centric tools because it is optimized to manage dimensions that change over time. In NICE PM, the tracking of such changes is referred to as *temporality*. This functionality is critical for performance management because measurement is fundamentally about people and groups, and the relationships between people and groups change frequently over time. The NICE platform offers features specifically designed to deal with temporality:

- **Temporal data model.** You can set temporal properties for any data that supports hierarchies (dimensional data).
- **Temporal identities.** Namespace identifiers from source systems can be temporal. For example, NICE PM can record that `Avaya 1234` belonged to Joe Smith before March 2006 and Jane Doe after March 2006.
- **Temporal attributes.** In addition to tracking the movement of entities such as agents or teams over time, you can also temporally track the related attributes of an entity. For example, Title might be an attribute related to the Person dimension. You can maintain a historical record of a person's job title.
- **Temporal report modes.** You can adjust both report labels and report data to reflect temporal changes, and you can show report labels for a selected time period. For example, if you choose "For all time" in a report showing people on Team A, the data shows any person ever assigned to Team A.

  You can also show data given a report's current constraints, or show an unconstrained view of the data. For example, if a report shows last month's data for people on Team A, and one of the agents moved to Team B midway through the month, you can constrain the report to show data only for the dates the agent was on Team A or unconstrain the report to show the agent's data for the entire month, even when he was on Team B.
- **Temporal filters.** NICE PM filters take into account changes in hierarchy relationships and attribute values over time. For example, suppose a report shows four months of data for each agent, and an agent was part-time for two months and full-time for two months. If a filter that excludes part-time employees is applied to the report, only two months of data appear for that agent.
- **Temporally precise calculations.** Using advanced metric expressions, administrators can precisely define how to allocate data to different entities in the presence of temporal motion. For example, if Joe changed teams during the report period, you can choose how to allocate his monthly score to team roll-ups for that period: average, first team, last team, and the like.
- **Temporal metric definitions.** Metric expressions can be made conditional on time ranges. For example, `if date < 3/06, then goal = a + b, else if goal = a + c`.

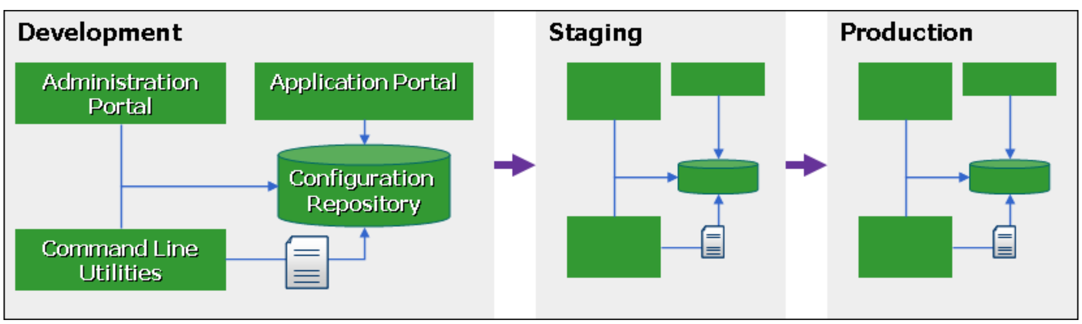
Configuration Framework

NICE PM applications are built on a highly flexible platform that enables administrators to customize the application. Application definitions are stored in metadata that in most cases can be modified using simple and intuitive browser-based authoring tools. The web authoring tools are available to users with administrative privileges on the Admin page. Configuration metadata manipulated through the authoring tools is stored directly in the database, and changes in application behavior are reflected immediately in the application without the need for server restarts.

Some types of configuration metadata, particularly metadata relating to data integration, are not stored in the database and is managed external to the user interface through a system of text-based property files.

The Configuration Framework is designed to support parallel development by multiple developers and a multi-stage rollout process typical for enterprise applications, including separate but parallel development, testing and production instances. Exchanging configurations between developers and across instances is facilitated by a powerful collection of import and export utilities that support conflict resolution, rollback, and modularization.

Figure 8: Configuration Framework



System administration

Because NICE PM already contains the most common performance management metrics and reports, the initial configuration can be brief and low impact for the customer. Configuration changes are done through intuitive browser-based GUIs that require no code to be written and can be used easily by business users such as analysts. With these tools, you can easily create and revise reports, metrics, forms, and alerts, and publish those configurations to the portal, applying personalization and access control settings for end users.

Involvement by highly skilled and often backlogged information technology (IT) staff can be minimized, and changes to the application can be rolled out to the user community more frequently. Because day-to-day configuration of the application is handled by business users, the primary responsibilities of IT staff are to:

- Maintain the database (backups, data purging) and hardware.
- Monitor data loads, log files, and network access problems.
- Add and revise data source integrations.

Access controls in the administrative tools ensure that only authorized users can access sensitive system configurations and data, such as sales or individual performance data.

NICE provides full documentation, training, and support for both IT and business user administration of the NICE product suite. For help modifying the application, contact your NICE representative.

Deployment

NICE PM is designed for ease and speed of installation and roll-out. It is a 100% browser-based application that can run on most operating systems and database platforms, making it easy to integrate into the existing IT environment. The GUI administration tools and simple end-user interfaces make rollout and training fast and easy for both IT and business users. Sophisticated administrative tools support parallel development and typical multi-stage rollouts from development to staging to production.

Reliability and scalability

NICE PM is designed to support deployments of tens of thousands of agents across many contact center sites. With an embedded J2EE application server and web server, the NICE platform is deployable to clustered environments that support enterprise scale load balancing and fail-over features. NICE PM makes effective use of data and metadata caching to guarantee high performance even at peak loads with large data sets. The NICE application is built to integrate with existing SNMP or JMS monitoring systems to notify of possible problems.

Where to get more information

The following table lists the guides that describe how to install, configure, and use the functionality described in this chapter. Additionally, see Chapter 4, "Getting Started with NICE PM," for tasks associated with your role.

For information on…	See…
Data mart and Data Integration Framework Org chart services Temporality services	*Data Management Guide*
Analytic Framework	*Cube Query Language Guide*
Transactions Framework Publishing Framework	*Application Configuration Guide*
Workflow Framework	*Forms and Workflow Configuration Guide*
Configuration Framework	*Cbean Configuration Guide*
Installation and configuration	*Installation Guide*
System administration	*System Administration Guide*
Pre-built content	*User Guide* and online help
Coaching forms and best practices library	*Coaching Plus Guide*
Using analytical tools to derive more value from your data and working offline	*Analyst Workbench Guide*
Using gamification to offer rewards and incentives to employees	*Engagement Plus Guide*
Configuring localization	*Global Edition Guide*

3 System Requirements

This chapter lists the hardware and software requirements for NICE PM. It contains the following sections:

NICE PM environments

NICE Systems recommends that you create the following separate, standalone environments for NICE PM:

- **Development**—A NICE PM environment used by your application developers to develop new extensions or new subsystems or to try out new configurations.
- **Test**—A NICE PM environment used by your application developers, system integrators, and system administrators to test new versions of the application or third-party software, extensions, or subsystems. Make sure this system has the same topology as your production environment so that testing is realistic.
- **Production**—A NICE PM environment in active use by end users. The performance and reliability of this environment is critical.

Each of these environments must have the required components, such as web servers, application servers, and a database. The goal of having separate environments is to allow your developers the freedom to create and your testers the freedom to stress the system, at the same time providing your end users the most stable environment possible.

Architectural requirements

The following sections describe the computer and network requirements in detail.

Web tier

The Web Tier manages web sessions and provides caching and load balancing across the Application Tier. It uses Apache HTTP Servers with the mod_proxy module for load balancing.

This tier runs either Microsoft Windows Server or Red Hat Linux. The processing requirements of this tier are low, so it can reside on shared hardware. A minimum of two computers are required for physical failover.

Application tier

The Application Tier runs the NICE PM application. NICE PM requires the Oracle Java Development Kit (JDK), which includes the required Java Runtime Environment (JRE) and is available for download free of charge.

This tier uses x86 hardware and Microsoft Windows Server or Red Hat Enterprise Linux. Run this tier on dedicated computers. A minimum of two computers are required for physical failover.

Data-load computer

The NICE PM data-load computer runs cube population jobs as well as the Data Integration and Cleansing Engine (DICE) and is part of the Application Tier. This computer has the same requirements as the rest of the Application Tier, including supported operating systems and prerequisite software. It runs batch jobs from a command line interface, typically initiated by scheduling software (such as Windows Scheduled Tasks, CONTROL-M, or cron).

The data-load computer can share the same physical hardware as the client-facing application instances; however, during the batch loading process, the client-facing application will have reduced capacity.

Data tier

The Data Tier hosts the relational database management system (RDBMS). NICE PM interacts with the Data Tier through two channels:

- Data load functionality (cubepop and DICE), which occurs on a scheduled batch basis and requires high throughput and insert performance, as well as internal optimization processing.
- Ongoing query and analytics functionality from users accessing NICE PM.

Data load functionality is expensive from a resource utilization perspective, and in larger environments the ongoing query functionality can be expensive as well. For this reason, NICE Systems strongly recommends a dedicated Data Tier. Data storage should have a minimum sustained throughput of 30 MB/sec.

Cluster requirements

When planning the number of server computers in a production cluster, consider the following requirements:

- A minimum of two application server computers for failover
- A minimum of one NICE PM server per 2000 licensed users (assuming 500 concurrent active users)
- A minimum of one NICE PM platform server per 5000 licensed users (assuming 1250 concurrent active users)
- All cluster components (nodes, hosts, application servers, and web servers) should be connected by a gigabit Ethernet network

NICE PM production and database server computers must meet the following minimum system configurations:

Computer	CPU	RAM
NICE PM servers	Standard server class, 4 cores (minimum 4 cores per computer, 2.66 GHz or higher)	8 GB
NICE PM Platform servers	Standard server class, 8 cores (minimum 8 cores per computer, 2.66 GHz or higher)	16 GB
Database servers	Standard server class, 4 cores (minimum 4 cores per computer, 2.66 GHz or higher)	16 GB

NOTE: The NICE PM implementation process includes customer-specific sizing.

Network recommendations

This section lists the network configuration recommendations.

Subnet configuration

Install the database and application servers within the same subnet. Database query packet sizes can be large, and traversing large distances across the network or through multiple subnets can significantly hinder performance.

Data encryption with SSL

Enable SSL (secure sockets layer) on web server computers. SSL uses a private key to encrypt data as it travels over the network and it requires you to obtain and maintain a valid digital server certificate from a trusted certificate authority. Certificates must be trusted by client web browsers, not by NICE PM.

NICE recommends that you obtain a certificate in PEM-encoded X.509 format. For more information on this format, see en.wikipedia.org/wiki/X.509.

Firewall configuration

NICE strongly recommends configuring a firewall in front of your NICE PM cluster and load balancer to allow incoming connections only to:

- The “front door” HTTPS port
- Any secure administration ports that are required

The firewall should disallow incoming connections to:

- Application servers
- Non-essential administration ports

The firewall must also allow outgoing connections for cubepop and DICE jobs, which use a variety of ports and protocols to connect with source systems.

The ports that you must open depend on the web server and NICE PM configurations.

Hardware sharing

In production environments, the components of a NICE PM cluster can run on different computers for optimal performance. When implementing NICE PM, consider the following:

- Web server instances (proxies) require little memory and can therefore share hardware with memory-intensive application instances.
- DICE and cubepop jobs are processor- and memory-intensive. You might want to run them on dedicated computers.

When sizing your environment, weigh supportability and failover considerations in addition to the guidelines provided earlier. For example, while reducing the number of instances might provide improved maintainability, it also means a larger proportion of capacity is lost if an individual instance fails.

Using a non-system user

NICE recommends installing and running NICE PM as a non-system user (a user other than Administrator on Windows or root on Linux).

Installing the application as a non-system user ensures that regular users can run the application. It also prevents anyone from using the application to access and alter key system files.

Operating systems

NICE PM runs on specific versions of Windows Server and Linux and requires you to use a certified operating system and Java version combination.

Operation system versions

Computers in the Web and Application Tiers must run one the following operating systems:

Vendor	Operating System
Microsoft	Windows Server 2008 and 2012 (64-bit, with 10 GB JVMs highly recommended; see the next section for details)
Red Hat	Enterprise Linux 5.x (64-bit)
	Enterprise Linux 6.x (64-Bit)

Computers in the Data Tier can use any operating system as long as it is supported by the RDBMS.

JVM memory allocation

Each NICE PM instance is a Java process that requires a large block of memory. A 64-bit Java Virtual Machine (JVM) is required. The bit width of the JVM is constrained by the bit width of your operating system.

Given its high addressable memory limit, a 64-bit JVM allows you to assign more memory to each NICE PM instance, which supports more concurrent users. You specify memory usage in the JVM configuration.

You can run multiple NICE PM instances on a single computer if sufficient resources are available for each deployed instance.

For more information about running NICE PM on 64-bit operating systems, see the *System Administration Guide*.

Virtualized environments

NICE supports running NICE PM on virtual servers, as long as the virtualized operating system is one of the supported versions. However, NICE cannot guarantee performance due to additional tuning variables in virtual environments. Ensure that any virtualized servers perform at the same level as the physical machines recommended in this document. Validate functionality in a virtualized test environment before deploying the application in your production environment.

NICE Systems reserves the right to request that any issues perceived to be virtualization-related are recreated in a non-virtualized environment.

Third-party software

NICE PM requires you to install specific software applications from third-party vendors.

Web servers

NICE PM web servers require the following third-party software application:

- **OpenSSL**—Apache HTTP Server uses SSL for encrypted communication. You must download and install OpenSSL v0.9.8n Light. For more information, see www.slproweb.com/products/Win32OpenSSL.html.
- **Microsoft Visual C++ Redistributables** (Windows only)—OpenSSL on Windows uses runtime libraries from the Microsoft Visual C++ 2008 Redistributable Package. You must download and install this package from Microsoft. For more information, see msdn.microsoft.com/library/ms235299.aspx.
- **Apache HTTP server—**Version 2.2.10 or later. In Linux environments using Red Hat Enterprise 6.x, the Apache server was likely installed with the distribution, or can be installed from the RPM Package Manager and configured using the Apache installer provided by NICE. If you are using RHEL 5.x, the version installed with the distribution is not compatible with NICE PM. Contact your NICE support representative for help installing and configuring the server. In you are using Windows, you must download and install the software separately. For more information, see httpd.apache.org/download.cgi.

Application servers

NICE PM application servers require the following third-party software applications:

- JDK 7.0 (also known as 1.7.0)—NICE recommends that you install Java version 1.7.0 update 10 or later. NICE PM runs on Oracle's Java Virtual Machine (JVM), included in the JDK and the JRE, and checks for JRE 1.7.0 update 10 or later.

  To download this version of the JDK, go to www.oracle.com/technetwork/java/archive-139210.html. Click the **Java SE 7** link. Scroll down to the Java SE Development Kit 7u10 section, and then download the version of the JDK that corresponds to your platform.

  You can still run NICE PM using some earlier versions of the JDK. For more information, see the *Release Guide*.

Database servers

NICE PM supports the following relational database management systems:

RDBMS	Versions
Microsoft SQL Server	2005, 2008, 2012
Oracle	10g, 11g

Unicode databases are supported for new product installations only.

Database driver Java archives (JARs) for the supported databases are embedded in NICE PM.

Clients

Clients must have the following third-party applications installed:

- Any of the following web browsers:
  - **Microsoft Internet Explorer 8, 9, or 10**—Use the 32-bit version of Internet Explorer, even if you are running a 64-bit OS. JavaScript must be enabled in all supported versions. No plug-ins are required, and Compatibility View is disabled by NICE PM automatically. The display resolution should be at least 1024x768 pixels. Pop-ups must be enabled to access the online help.
  - **Mozilla Firefox 21 or later**—JavaScript must be enabled. No plug-ins are required. The display resolution should be at least 1024x768 pixels. Pop-ups must be enabled to access the online help.
  - **Apple Safari 6.0.4 or later**—You can run Safari on either OS X or iOS.
- **Adobe Acrobat or Acrobat Reader 10.0** or later—Required for the Create PDF feature.
- **Microsoft Excel 2007 or 2010**—Required for the Export to Excel feature and the Merced XL plug-in for the Analyst Workbench module.

NOTE: NICE Systems certifies only Windows and Internet Explorer, even though NICE PM might run on other operating systems and browsers.

Mobile clients

Users can access the application on the following mobile devices:

- Apple iPads running Safari on iOS 6
- Samsung Galaxy tablets running Mozilla Firefox for mobile or Google Chrome for Android on Android 4.1.1

Single sign-on authentication

You can integrate the following single sign-on authentication solutions with NICE PM. If you want to use another solution, contact your NICE representative.

- NTLMv1
- SiteMinder
- SPNEGO
- LDAP
- SAML 2.0
- AES/Discover
- Salesforce.com
- Encryption-based

4 Getting Started with NICE PM

This chapter describes the common divisions of responsibility among users responsible for the NICE PM system and the tasks they most frequently need to perform. It also describes where to find information on how to perform the tasks in the online help and documentation.

Depending on the size and complexity of your NICE PM implementation, a few people might be responsible for several roles.

This chapter describes the following topics:

PMCC owner

The Performance Management Center of Competency (PMCC) owner is a group that plans and executes the NICE PM implementation. The group defines the overall strategy, which includes selecting metrics, modeling behavior analysis and outlier management, and defining a coaching process. Then they gather feedback and inputs, generate ideas, and execute the implementation.

Typically the PMCC owner works with the Business owner to make decisions on core features or functionality pertaining to business requirements that need to be included in the system.

For information about NICE PM, see the previous chapters in this guide.

Business owner

Typically the business owner, along with the PMCC owner, makes decisions on core features and functionality pertaining to business requirements that need to be included in the system. The business owner is typically not involved in the day-to-day project.

For information about NICE PM, see the previous chapters in this guide.

Analyst

Analysts perform several functions in NICE PM, such as analyzing group performance, organizational KPI performance, overall trends, organizational adoption, business impact, coaching effectiveness, and talent management.

Analysts are also responsible for managing the content and data integrity of NICE PM. Tasks include designing and documenting the organization roll up and drill down functionality; business metrics and associated logic; report, dashboard and form layouts; access control and data restriction policies; and data validation.

For information about analyst tasks, see “Getting Started” in the online help and see the chapters on configuring reports in the *Application Configuration Guide*. For information about using NICE XL to run reports in Microsoft Excel, see the *Analyst Workbench Guide*.

Operations analyst

The operations analyst makes sure the scheduled jobs run and analyzes the resulting operational metrics on a daily or more frequent basis.

For more information about operations analyst tasks, see “Getting Started” in the online help, and also see the *Data Management Guide*.

Designer

The designer translates business definitions into NICE PM designs. Typically the person in this role is a business analyst with strong technical aptitude or an IT resource with strong business skills.

For information on a designer’s tasks, see “Getting Started” in the online help, and the *Application Configuration Guide*.

Frontline support technician

The frontline support technician is a system administration role that helps users with lost passwords and answers task-related questions.

For information on these tasks, see the chapter on authentication in the *System Administration Guide*, and see "Getting Started" in the online help.

Back-end support technician

The back-end support technician is a system administration role that tracks system health, including the NICE PM servers, database, data jobs, and so on.

For information on these tasks, see Chapter 3, "System Requirements," and the *System Administration Guide*. Also see "Getting Started" in the online help.

Back-end developer

The back-end developer is a system administration role that handles content configuration, such as data models, DICE jobs, forms and workflow, cluster configuration, and application branding. This role is similar to the Data source owner role.

Back-end configuration includes all activities involving properties files, which requires knowledge of how to work with the servers, run commands from the command line, review log files, and so on.

For information on back-end developer tasks, see Chapter 3, "System Requirements," and the *System Administration Guide*. Also see "Getting Started" in the online help.

Front-end developer

The front-end developer is responsible for content configuration: reports, forms, workflows, dashboards, metrics, OLAP alerts, and so on.

Front-end configuration involves cbeans and anything that can be configured using the Admin user interface, such as reports, dashboards, metrics, and so on. Front-end developers will often get requirements and design recommendations from analysts. Staff in this role do not need to know how to re-start the system or work with different servers.

For information on front-end developer tasks, see the *Application Configuration Guide*, the *Cbean Configuration Guide*, the *Forms and Workflow Configuration Guide*, the *Data Management Guide*, and (optional) the *Analyst Workbench Guide*. Also see "Getting Started" in the online help.

System project manager

The system project manager is a system administration role that involves project management—what gets built, who builds it, when it is built, and how it is released. This is also the application owner.

For information on system project manager tasks, see the *Release Guide*, the *Application Configuration Guide*, the *Data Management Guide*, and the pre-deployment checklist in the *System Administration Guide*. Also see "Getting Started" in the online help.

Data source owner

Similar to the Back-end developer role, the data source owners are responsible for assisting the core project team in data source discovery and data load design, providing SQL queries to build data conversions, and access to source data systems. Data source owners also typically participate in documentation efforts.

For information on data source owner tasks, see the *Data Management Guide*. Also see "Getting Started" in the online help.

Additional support roles

As with any business-critical system, the NICE PM database needs to be monitored and managed by an experienced Database Administrator, and its connections with external systems should be configured and maintained by a System and Network Administrator.

For more information, see the *System Administration Guide*. Also see "Getting Started" in the online help.

Managers and supervisors

Some of the manager roles that use NICE PM include QM managers, communications managers, release managers, training managers, and supervisors. People in these roles typically perform post-implementation tasks, such as using transactional reports and searches, recognizing employees, and filling out coaching and quality forms.

For more information, see the online help topics on using reports, dashboards, and forms. For a list of terms used in NICE PM, see the Glossary in this guide.

Agents

Before using NICE PM for the first time, you need to know a few basics that most agents and other users typically perform, such as how to log in, and how to use the home page and navigate to many features.

For information on these tasks, see “Getting Started” in the online help.

Glossary

The glossary defines terms used in the NICE PM documentation and common terms that have a specific meaning in NICE PM.

A

absolute phrases

Phrases used to define specific sets of axis coordinates that remain the same regardless of the logged-in user.

See also **phrases** and **relative phrases**.

agent group

A special group that holds only employees with the agent title.

See also **groups**.

aggregator

A function that performs an aggregation on a set of data.

alerts

System-generated notifications sent to users to warn them about specific metric thresholds, tell them about events, or remind them about tasks. Alerts are associated with events or conditions, which generate the alert.

See also **conditions, for alerts** and **fields, in alert messages**.

application portal

Controls the structure and display of the NICE PM user interface. The portal is where administrators create pages and publish content items, including reports and dashboards, and assign their view permissions.

attributes

Properties of hierarchies. An attribute is a "shorthand" mechanism for defining a one-level hierarchy. For example, a Function attribute can be defined for the Team grain, or a Primary Skill attribute can be defined for the Person grain.

automatic cube management

The process in which CubePop automatically creates cubes from definitions of base metrics and assigns base metrics to appropriate cubes. NICE PM provides automatic cube management to allow companies to manage their application without having to define cubes.

See also **CubePop** and **NextPop**.

axes

Specific instances of dimensions added to reports. For example, the Time dimension can be added as an axis to a report.

The x-axis is the column axis of the report. The y-axis is the row axis. The z-axes appear above the report and are fixed, or held constant.

See also **dimensions**.

B

base jobs

Data jobs defined to appear on the Data Status and View Job Status screens in NICE PM. Usually base jobs are data jobs that the user initiates or schedules.

See also **data jobs**.

base metric cube

Cubes filled with data that did not result from a calculated metric expression.

base metrics

Metrics that provide numerical data derived from specified metric sources (facts). Base metrics represent data collected and stored in the database.

A base metric configuration references a metric source, specifies a fact source, and sets the method by which the data will be aggregated.

See also facts and **metric sources**.

breadcrumb

A navigational element that appears across the top of the NICE PM screen to help users track their location within the application.

C

calculated metrics

Metrics derived from a combination of any number of base or calculated metrics with any number of operations, constants, or both.

See also **base metrics**.

cbeans

Objects used in NICE PM to store and organize configurations and contents to allow for easy authoring, sharing, and exporting. Cbeans are stored in the cbean repository.

See also **metabeans**.

Cbuild

The tool used to compile or build configurations into a NICE PM application.

challenge

A time-based pursuit that uses one or more metrics to determine whether the required objectives are met in order to complete the pursuit.

See also gamification and **pursuit**.

charts

Visual representations of the data in reports.

chunks

Tables in the NICE PM data mart.

See also **metabean type**.

compound report

Two separate reports placed close together in a dashboard row or column and configured so that they appear as a single report in the dashboard.

conditions, for alerts

Expressions that define the values and measures of a metric that should trigger an alert. An alert can have one or many conditions.

See also **alerts**.

conditions, for filters

Expressions comparing dimensional entities, that create filters.

See also **filters**.

configurations

Objects in NICE PM that determine how the application behaves. To configure NICE PM, you create configurations of various types that model your application. Each category of items is a different configuration *type* and each configuration type can contain many individual properties.

configuration dependencies

References from a configuration to other configurations in NICE PM. Many configurations reference other existing configurations. For example, a dashboard can reference a report, a transactional report, and alerts. In this example, the report, transactional report, and alerts are the dependencies.

See also **configurations**.

Configuration Manager

A portlet that allows you to create, search for, view, edit, copy, delete, and export cbeans. You can also use it to view the relationships between cbeans and group them into related modules.

Configuration Project Manager

A tool that merges configurations from multiple instances of a NICE PM application. This allows developers to work simultaneously and independently and then merge their work without conflicts. It is similar to a version control system such as CVS, but with functionality specific to NICE PM cbean configurations.

constraint grain

The grain specification within a specializer that specifies which grain the specializer applies to.

content items

Configuration types that are publishable. Content items are available to users when they are published in the portal.

See also **configurations**.

content tree

The tool you use to publish content items for mobile applications and other similar deployments that reference only the NICE PM platform instead of the Liferay portal. The content tree lists the main tabs (such as `home` for the home page or `actions` for the Action page), their subdirectories (such as `landing-page`), and the content items contained in each subdirectory.

coordinate

An ordered set of points.

coordinate expression

An expression specifying a dimension, a point grain, and selection criteria that will generate a set of points when evaluated by the analytics engine.

cube aggregates

Sibling cubes pre-aggregated to coarser grains. Cube aggregates can improve report performance. If a user rolls up data in a report and an aggregate cube exists, NICE PM doesn't have to aggregate metrics to render the report; instead, the report can simply query the cube aggregate.

See also **cubes** and **sibling architecture**.

CubePop

A tool that creates or updates data in every existing dimension table and fact table. Fact tables that CubePop loads reference data in newly-created or updated dimension tables.

Cube Query Language

The language used to query cubes for data. Application developers use functions such as `numeric operators` or `balanced scores` to define cube functions that analysts or administrators need when developing the reports used in NICE PM.

cubes

A collection of metrics described by data that play specific roles in conjunction with those metrics. Also, a logical organization of multidimensional data.

See also **edge**.

customer layer

One of three application layers, the customer layer contains customized configuration files that override the application's default behavior. Folders in this layer can include `apps`, `branding`, `common`, `models`, `phrases`, and `src`. The other application layers include the foundation layer and the platform layer.

See also **properties files**.

D

dashboards

Content items that enable combining multiple content items on a single page. Typically, reports contained in a dashboard address a similar theme. Other items that dashboards can contain are: transactional reports, free form text items, messages, alerts, and tasks.

database statistics

Information that helps the DBMS determine how to best execute queries. This information includes details about the size and physical layout of tables, columns, and indexes. Statistics count the number of rows, record the blocks of data on the disk, and measure whether the data is dense or fragmented.

See also **DBMS**.

data jobs

Jobs that populate the data mart.

See also **jobs**.

data mart

A set of relational database tables optimized for analysis and operational or transactional use. CubePop populates the data mart with data that DICE extracts from external company sources and data entered into forms. The data mart organizes data into fact tables (cubes) and dimension tables. Most end-user modules use the fact and dimension tables in the data mart to create reports and dashboards.

data restriction policy

A list of titles and a data restriction relationship for each title. The restriction relationship indicates whose data can be accessed by persons with that title. For example, a data restriction policy might give directors access to information about everyone, managers access to information about their team members, and agents no access at all.

See also **titles**.

DBMS

Database management system. A software system that builds and maintains database tables.

DICE

Data Integration and Cleansing Engine. DICE is the process that extracts data from data sources and loads the data into the NICE PM data mart. DICE is an ETL (extract, transform, and load) engine.

DICE conversions

Specifications for extracting, loading, and transforming data from source systems into the data mart.

See also **DICE**.

DICE jobs

Sets of DICE conversions and NextPop populations run together, either in sequence or in parallel.

See also **DICE**, **jobs**, and **data jobs**.

dimensions

Spaces that an organization uses to measure values, such as the time space. Dimensions correspond to axes.

dimension labels

Specifications of dimensions on a report's edges. Labels describe a point or area in the dimension.

See also **dimensions**.

dimension tables

Database tables that store (in the OLAP repository) all possible values for a particular dimension.

See also **dimensions**.

drilling down

The process of following a dimension's predefined drill path to the next finer grain. For example, in a report that displays months, the user can drill down the time dimension to view the report data in days.

See also **rolling up** and **drill paths**.

drill paths

Configurations that define the grains used for the drill down and roll up feature and the quick drill feature in reports. Drill paths allow administrators to control how users drill down to view items at a finer data view or roll up to view items at a coarser data view.

See also **drilling down** and **rolling up**.

dynamic link

A hyperlink, available in a source item's user interface, that leads to a target resource and can be configured to present the target resource in something other than its default state. For example, the target resource could be based on the state of the source item. A dynamic link is represented internally as a cbean, within the definition of the source item's cbean.

See also **source item** and **target resource**.

dynamic role

A role available in the portal that leverages profiles.

See also **role** and **profiles**.

E

edge

A side of a cube where information describing the cube's facts is located, or the resulting set of points of a dimension query. Data is often organized along a row edge, a column edge, and a page edge, though there is no limit to the number of edges on a cube.

See also **cubes**.

entity relationship diagram (ERD)

A data modeling technique that creates a graphical representation of the entities, and the relationships between entities, within an information system.

external links

Content items containing a link to any web page outside (or inside) of the application. After an external link is created, it must be published to a menu or page for users to view and use it.

See also **publishing**.

F

fact source

The column within a fact table that contains the facts for a metric.

fact tables

Database tables that store (in the OLAP repository) values measured for and associated with dimension tables. Cubes are made up of fact tables. In NICE PM, you configure fact tables with base metrics.

See also **base metrics**.

facts

The data points recorded that can be measured, typically the data points inside a cube.

fettered

To be weighed down by constraints.

fields, in alert messages

Placeholder items that insert text, a link, or a value into the subject or body of a message when a message is generated for an alert.

See also **alerts**.

filters

Pre-defined constraints that allow users to narrow the focus of the report. For example, the user could choose to exclude supervisors from a report, to display only agent data. Typically filters are defined for each dimension, such as Time filters and Person filters.

See also **conditions, for filters**.

forms

Screens in NICE PM that let users enter data directly into the data mart. NICE PM then integrates this form data with the other data in OLAP and transactional reports.

form presenters

Form presenters control the type and appearance of a field. They allow the person setting up a NICE PM instance to create a field as a text box, a check box, a drop-down list, a date presenter, or another format.

form validators

Validators are cbean configurations assigned to particular fields on a form to validate whether information entered in that field meets required conditions. When a form is submitted, the validators perform basic checks on the data entered in their given fields, testing that it satisfies specific constraints before saving the form. For example, validators can check whether required fields contain data, whether text exceeds maximum character lengths, or that number fields are within a given range.

free form text

Content items that allow the insertion of text blocks or other HTML formatted content in dashboards or on their own in NICE PM. Free form text items can be created to contain any kind of text such as welcome messages in dashboards or informational messages on the home page or other landing screens.

G

gamification

A tool that applies the mechanics of gaming to non-game activities to change people’s behavior. In a business context, gamification is the process of integrating game dynamics and game mechanics into an application in order to improve employee performance and drive participation and engagement.

See also **pursuit**.

grains

An entity at a specific level of detail. A dimension consists of a series of related grains.

See also **hierarchy**.

granularity

A specific level of detail. A finer grain represents more detail, while a coarser grain represents less detail, or a roll-up of finer detail.

groups

Particular employees or collection of employees who work at specific levels and have specific functions in an organization’s hierarchy.

See also **group types**.

group types

Hierarchical levels within a company's organization. Each group is assigned a group type. The group types are configured per application, but may typically include levels such as: Organization, Site, Department, and Team.

See also **groups** and **agent group**.

H

hierarchy

A structure that describes the leveled relationships between entity types as those entities are structured in the data mart.

See also **hierarchy entity**, **dimensions**, and **grains**.

hierarchy conversion

The task used to load hierarchy data from the org chart into NICE PM.

See also **hierarchy**.

hierarchy entity

An instance of an item modeled in a hierarchy, such as a person in the Person hierarchy, a specific product in the Product hierarchy, or a split in the Split hierarchy.

See also **dimensions**.

Hierarchy Manager

A tool available on the Admin portlet that allows you to create and maintain org chart data for NICE PM. Specifically, the Hierarchy Manager allows you to manage employees and groups, including changing team assignments, initiating or terminating employees, creating new groups and employees, reconciling multiple IDs inherited from multiple systems, and reconciling "phantom" IDs.

history ("hist") tables

Tables that contain historical or temporal associations between grains, attributes, or both.

I, J

identities

Person identifiers. NICE PM maintains a unique NICE ID for each person imported from the org chart. NICE PM links individual persons or groups and their unique NICE IDs to specific identifiers in various source systems, such as WFM and ACD. The source system identifier is called the namespace identifier.

JDBC

Java database connectivity. The part of the Java Development Kit that defines an application programming interface to give Java programs standard SQL access to databases.

job components

Components that define a task or set of tasks to complete. A job component might initiate other job components, called child job components, to complete some tasks. In NICE PM, properties files often define job components.

See also **jobs**.

jobs

Job executions, executed by job components. A job component defines a task or set of tasks to complete. A job component can initiate other job components, called child job components, to complete some tasks.

See also **base jobs**, **data jobs**, and **job components**.

L

landing page

The location of items displayed on the main screen while using the portal. For example, to place a free form text item on the main screen displayed when users log in, place it on the page named `home` in the portal.

levels

The pieces that make up dimensions.

See also **dimensions**.

M

MERCED_HOME

The single directory on the NICE server that contains the files and executables that support the application, including application and web servers as well as jobs. You can give the actual directory a different name and create a system variable called `MERCED_HOME` that references the actual directory.

messages

User-generated notifications that provide a way to communicate inside of NICE PM. Alerts can also be configured to send messages when their conditions are met.

See also **alerts**.

metabeans

A metabean corresponds to a row in a table and has constituent properties. Those properties are specified by another object, a metabean type.

See also **metabean type**.

metabean type

An object that specifies unique properties that describe a particular metabean. A metabean type corresponds to a table, as a metabean corresponds to a row in the table. NICE PM uses metabean types to create tables.

See also **metabeans**.

method of aggregation

The method used when aggregating data into a higher grain. Explicit, sum, min, or max.

metric functions

Higher-order business functions that execute complex expressions. Metric functions use Cube Query Language (CQL), which is used in NICE PM to define calculated metrics. For more information, see the *Cube Query Language Guide*.

See also **calculated metrics**.

metric sources

Data objects that describes data in specific data mart chunk tables, enabling users to access subsets of that data without the need for query languages. Once configured, metric sources can be used by any number of base metric configurations, which in turn form the building blocks for more complex calculated and status metrics.

metrics

Data points you want to measure that are derived from the aggregation of facts. There are three types of metrics in NICE PM: base metrics, calculated metrics, and status metrics.

See also **base metrics**, **calculated metrics**, and **status metrics**.

modules

Collections of configurations and content items that define a logical unit of functionality. Modules are useful to help organize configurations and content items in NICE PM.

See also **configurations**.

multi-level metrics

Metrics that rely on data aggregated from more than one grain in a hierarchy.

N

NextPop

A process that updates an individual fact table in the OLAP engine with fresh data from the data mart. Fact tables that NextPop loads continue to reference current dimension tables. You run NextPop from a DICE job.

See also **cubes** and **CubePop**.

normalize

To organize data and break it into smaller tables that are easier to manage. Helps avoid having to maintain redundant data.

O

objectives

The conditions that must be satisfied in order to win or complete a pursuit.

See also gamification and **pursuit**.

ODBC

Open database connectivity. A standard for accessing different database systems. An application can submit statements to ODBC using the ODBC flavor of SQL. ODBC translates these statements to whatever flavor the database understands.

OLAP

Online analytical processing. A data model, used for complex analysis, that thinks of data as locations in a multi-dimensional space.

OLTP

Online transaction processing.

org chart

The persons and groups, and the relationships between the persons and groups, that NICE PM stores data for and reports on.

See also **person** and **groups**.

P

person

An employee in the org chart. Each person must have a unique identifier, title, and parent group, in addition to characteristics like name.

See also **org chart**, **titles**, and **groups**.

person phrase

A person phrase that shows various reporting structures in the Person dimension. Some examples include **My Teams**, **People Below Me**, and **Agents on Team Reggie Putnam**.

See also **relative phrases**, **absolute phrases** and **time phrase**.

personalization

Controls that determine what content users see when they log in to NICE PM. Personalization allows one or more sets of permissions, called profiles, to be applied to content in NICE PM. Once applied, only users who fit the profile criteria will be able to view or access the content. Using personalization, permissions can be applied to individual content items or, through the portal, entire sub-directories.

See also **profiles**.

phantom IDs

IDs that NICE PM creates when the system contains data about events before the person active in those events exists in NICE PM. With no *real* ID to attach to the record, NICE PM creates and assigns a phantom ID instead. Phantom IDs can then be handled and resolved by an administrator.

phrases

Phrases are used to define the coordinates to be displayed on each axis in a report. A phrase is a pre-defined instance of an axis. For example, "This month" could be a phrase defined for the Time dimension axis.

See also **axes**, **relative phrases**, **absolute phrases** and **time phrase**.

platform

Provides the common services used by the system, such as the platform core, platform service implementation, and platform notification services, and contains the service platform. The platform database is used for the platform services, such as user management.

See also **service platform**.

point

An entity within a dimension.

point grain

The granularity to which facts are to be aggregated.

point-wise operations

Operations evaluated in a fixed frame and applied across corresponding points.

portlets

Content items that display application data on pages in the portal.

private pages

One of the two page types in Liferay, and the only page type you can create in NICE PM. All of the pages in the application are private except for the login screen. Private pages are available only to authenticated users.

profiles

Sets of criteria used to group users based on attributes (for example, titles) and other attributes that make possible more refined groupings. For example, agents that speak Spanish or supervisors with additional administrative privileges.

Also used to describe customizable pages about users that contain basic information (such as name, title, and department), a brief biographical statement, activity feed, distinctions, current quests, and links to other users in the organization.

properties files

Files used, along with cbeans, to define and configure NICE PM components. A properties file is a text file with a `.properties` extension. It contains name/value assignments of the form *name=value*, and generally lists one assignment per line.

See also **cbeans** and **customer layer**.

publishing

Using the portal or content tree to post an item or a link to a content item on a screen, allowing users to access the item.

See also content tree.

pursuit

In Engagement Plus, a pursuit is an activity that results in the employee achieving a reward on successful completion. Pursuits are created by business administrators and used to encourage certain behaviors or improvements in employees who participate in them. Employees can be assigned a pursuit or join voluntarily. Some pursuits provide aggregated data that be used in reports and can show the employee's progress toward meeting a goal.

See also gamification.

Q, R

quest

A pursuit type in Engagement Plus. A structured series of objectives that result in a reward upon completion.

See also **pursuit** and gamification.

related metrics

Any existing metrics that report creators can make available through a current metric when it is used in dashboards and reports. Users can drill down into related metrics that were specified when the base metric was created.

relative phrases

Phrase used to define specific sets of axis coordinates that change depending on who is logged-in as the user. For example, “My Team.”

See also **phrases** and **absolute phrases**.

reporting frame

The frame defined by a report.

reporting grain

The granularity at which you want to perceive your data. (See **point grain**.)

reports

A view of a cube query, or a presentation of related metrics that allow users to understand and analyze data.

result cube

The cube formed as the result of evaluating a cube query.

ROLAP

Relational-OLAP. An architecture used in NICE PM that stores data in relational database tables.

See also **OLAP**.

role

A convention used to describe an entity’s involvement within a transaction. Each entity is a member of a dimension or a dimension member.

See also **dynamic role**.

role-playing dimension

An advanced configuration technique that allows NICE PM administrators to model more than one value from the same dimension in a single report. This allows users to compare values within the same dimension. For example, you can compare two Time dimensions such as when an incident happened, called **When**, and when the supervisor coached the agent about it, called **When Discussed**.

rolling up

The process of following a dimension's predefined drill path to the next courser grain. For example, in a report that displays months, the user can roll up the time dimension to view the report data in quarters.

See also **drilling down** and **drill paths**.

S

schema generator

The process NICE PM uses to create the tables in the data mart. This process, named `schemaload`, takes each registered metabean type and creates a database table in the format required by the DBMS.

selection criteria

The portion of a coordinate expression that specifies which points to select.

service platform

Part of the platform composed of the Enterprise Service Bus (ESB), which carries data between services and the rest of the application, and the services that provide user management, authentication, configuration, and other functionality.

See also **platform**.

shape-change operations

Operations that change the shape of cubes.

sibling architecture

The mechanism used to keep NICE PM available for reporting while CubePop online and NextPop populations are running. During populations, data is loaded into a copy, also called a sibling, of the table so that the original table is not effected and can continue to be reported on. After the data is loaded successfully, the copy is deployed, making it the active table that NICE PM uses for reporting.

See also **CubePop** and **NextPop**.

slave application

Any NICE application that synchronizes users to the service platform, such as NICE ICM and NICE PM.

See also **service platform**.

source item

A particular content item whose type is a Source Item Type.

source item type

The content type that can show dynamic links in its user interface. Currently source item types are OLAP reports and dashboards.

source system

Outside, third-party applications that contain data that is loaded into NICE PM. Also called the system of record.

specializer [expression]

The literal form used to specify selection criteria.

status metrics

Metrics that are formulated to evaluate expressions and display graphical icons that corresponds with the result. In this way, a status metric shows the “status” indicated by the value.

T

target resource

A particular resource whose type is a target resource type.

target resource type

A type of resource that can be the target (destination) of an dynamic link. The current target resource types are OLAP reports, dashboards, MPS 3.x forms, and workflows.

See also **dynamic link**.

tasks

User-generated notifications that require the completion of an action, such as completing a form. Tasks contain a due date and a status that indicates whether they are complete or not.

temporal extent mode

Specifies the temporal constraints used to determine whether to include an entity via the selection of dimension members.

temporal identities

Unique identities given to data in NICE PM that provide the capability for identities to be reassigned and re-used over time.

See also **temporality** and **identities**.

temporal instance

An entity with a fixed set of attributes for a given period of time.

temporality

The ability to track changes over time in relationships of dimension entities, such as people or groups. For example, tracking the movement of agents between teams.

temporal modes

A feature in NICE PM reports that lets users specify how to display data of entities that move over time. The different temporal modes the users can choose from are:

- Default—Uses the current temporal constraints of the axis's dimension. For example, a report might show data for agents on a specific team during a given time.
- Review—Relaxes the temporal constraints. For example, a report might show data for agents regardless of team during a given time.
- Custom—Displays a dialog box where the user can customize the temporal mode.

temporal relationship

A relationship between entities that changes over time.

Time dimension

The dimension in NICE PM that defines time.

See also **dimensions**.

time grain, custom

Time dimension grains that can be tailored to match the organization's model of how time works. For example, some organizations have unique definitions of the boundaries of fiscal years or fiscal months that don't match the standard time grains in NICE PM.

time phrase

Time phrases let you view historical, forecast, or current data. Some common examples are **Last Week**, **Last Month**, **Last Year,** and **Last Fiscal Quarter.**

See also **relative phrases**, **absolute phrases**, and **person phrase**.

time range, in reports

The time period displayed in a report. Users can click **Change** in the report's filter bar to change the time range.

titles

Employee types, usually related to employee function and level in the organization.

transactional data

Unaggregated (raw) data recorded about transactions or events that occurred.

transactional reports

Sets of data retrieved from tables in the data mart, based on matching pre-configured criteria.

transactional searches

Sets of data retrieved from tables in the data mart, based on matching user-entered criteria at retrieval time.

type-specific options

Dynamic link configuration options that are available, or configurable, only for a particular combination of source to target resource types. These options generally translate to actions. For example, report to report, report to dashboard, report to workflow, dashboard to dashboard, and dashboard to workflow.

See also **dynamic link**.

type-specific mapping

Type-specific options commonly include (or consist entirely of) some kind of mapping specifying which values in the source item will populate which fields in the target resource. The exact meaning of "value," "field," and "populate" varies, but these type-specific options can be referred to more specifically as the type-specific mapping, or just mapping when the context is clear. For example, when linking from a report to a dashboard, there is a mapping of report edges to dashboard member prompts.

See also **dynamic link**.

U

unfetter

To remove excessive constraints, such as temporal constraints.

units of measure (unit)

Elements that define the formats that NICE PM uses when displaying metrics. When authoring base and calculated metrics, the units that the metric results are expressed in, such as percentage, are specified by units of measure.

W

web server

An instance of a web server. When end-users request the NICE PM application in their browser, a web server determines what application server to run their browser sessions on, and sends browser requests to that server.

workflows

Tools for managing business processes in NICE PM. A workflow consists of forms, alerts, messages, and tasks. Workflows help a company manage processes related to customers, administration, or performance. For example, a workflow might help maintain customer satisfaction by initiating and tracking follow-up calls (using messages and tasks), maintain employee coaching (using forms), or warn automatically of performance issues (using alerts).

Y

yardsticks

Statistical metrics that aggregate or summarize data across sets of rows or columns. Adding a yardstick to the column or row axis of a report (or both), creates additional context for the report's metrics. For example, a yardstick can display the average number of interactions for teams within a department.

See also yardstick, external.

yardstick, external

Yardsticks that reference data outside of the report table—for example, comparing the agents' performance in the report to the performance of the site as a whole. External yardsticks can be used to provide benchmarks to compare data in reports to non-report data.

See also yardsticks.

Index