FUJITSU Software
Interstage Business Process Manager Analytics V12.2

Overview

Windows/Linux

About this Manual

This section explains summary, chapter overview, abbreviations, and provides useful information.

The contents in this section are as follows:

- Purpose

- Intended Audience

- Chapter Overview

- Typographical Conventions

- Reference Materials

- Abbreviations

- Export Regulations

- Trademarks

- Requests

- Copyright

Purpose

This user guide describes the architecture and features of the Interstage Business Process Manager Analytics (BPMA).

Intended Audience

This manual is intended for System Administrators. It is assumed that the end user has a working knowledge of the following topics:

- Operating system administration

- Application server administration

- Database administration

Chapter Overview

Chapter	Title	Description
1	Overview of Analytics	Basic concepts of Interstage BPM Analytics.
2	What Analytics Does	Descriptions of architecture and features.
3	Setting up and Configuring Analytics	An overview of the set-up sequence and configuration process.
4	Overview of BAM functionalities and Some Examples	Provides contents of getting started with BAM functionalities.
5	Some Examples with Process Discovery	Provides contents of getting started with Process Discovery.

Typographical Conventions

The following conventions are used throughout this manual.

Example	Meaning
Command	User input that must be typed is identified by the Courier font
screen text	- GUI options to be selected by the end user are **bold** - GUI options that are emphasized are **bold**
Reference	Reference material or related documentation is specified in *italics*

Example	Meaning
Parameter	Command parameters are identified by the Courier font

Reference Materials

The following related documentation is also available.

Release Notes

Contains an overview of the Analytics software and late-breaking information that could not be included in the manuals

Installation Guide

Explains how to install Analytics

Dashboard / Output Guide

Explains how to use the Analytics Dashboard

Administration Guide

Covers administration tools and tips, message references, and troubleshooting

Analytics Studio Guide

Explains how to use the Analytics Studio to configure parameters that enforce Analytics features

Management Console Guide

Explains how to use the Management Console and Management Commands to configure key parameters, such as how to start/stop the Analytics software service or how to display the software status

Process Discovery Guide

Explains how to use Process Discovery that generates process diagrams using historical data

Migration Guide

Explains how to migrate from previous version of the product

Handbook for the Process Discovery

Explains how to operate the Process Discovery briefly

Abbreviations

The following references for BPM Analytics are also available:

Name	Abbreviation
Microsoft(R) Windows Server(R) 2008 SP2 Enterprise Edition Microsoft(R) Windows Server(R) 2008 SP2 Standard Edition Microsoft(R) Windows Server(R) 2008 R2 Enterprise Microsoft(R) Windows Server(R) 2008 R2 Standard	Windows Server 2008, Windows Server
Microsoft(R) Windows Server(R) 2012 Datacenter Microsoft(R) Windows Server(R) 2012 Standard Microsoft(R) Windows Server(R) 2012 R2 Datacenter Microsoft(R) Windows Server(R) 2012 R2 Standard	Windows Server 2012, Windows Server
Microsoft(R) Windows(R) XP SP3 Professional operating system Microsoft(R) Windows(R) XP SP3 Home Edition operating system	Windows XP, Windows
Microsoft(R) Windows Vista(R) SP2 Ultimate Microsoft(R) Windows Vista(R) SP2 Enterprise Microsoft(R) Windows Vista(R) SP2 Business	Windows Vista, Windows

Name	Abbreviation
Microsoft(R) Windows(R) 7 Ultimate Microsoft(R) Windows(R) 7 Professional Microsoft(R) Windows(R) 7 Home Premium	Windows 7, Windows
Microsoft(R) Windows(R) 8 Enterprise Microsoft(R) Windows(R) 8 Pro Microsoft(R) Windows(R) 8	Windows 8, Windows
Microsoft(R) Windows(R) 8.1 Enterprise Microsoft(R) Windows(R) 8.1 Pro Microsoft(R) Windows(R) 8.1	Windows 8.1, Windows
Microsoft(R) Internet Explorer 7.0 Microsoft(R) Internet Explorer 8.0 Microsoft(R) Internet Explorer 9.0 Microsoft(R) Internet Explorer 10.0 Microsoft(R) Internet Explorer 11.0	Internet Explorer
Adobe Flash Player V11.3	Flash Player
Red Hat Enterprise Linux 5 Red Hat Enterprise Linux 6	Linux
Android 4.0.4	Android
i-OS6	iOS
Interstage Application Server Enterprise Edition Interstage Application Server Standard-J Edition	Interstage Application Server
Oracle WebLogic Server 12c	WebLogic
WebSphere Application Server Network Deployment V8.5	WebSphere
RedHat JBOSS Enterprise Application Platform 5.1.1	JBoss Enterprise Application Platform 5.1, JBoss
RedHat JBOSS Enterprise Application Platform 6.1.1	JBoss Enterprise Application Platform 6.1, JBoss
Interstage Business Process Manager	Interstage BPM
Interstage Service Integrator	ISI
Oracle Database 10g R2 Enterprise Edition Oracle Database 10g R2 Standard Edition Oracle Database 10g R2 Standard Edition One	Oracle 10g, Oracle
Oracle Database 11g R2 Enterprise Edition Oracle Database 11g R2 Standard Edition Oracle Database 11g R2 Standard Edition One	Oracle 11g, Oracle
Oracle Database 12c R1 Enterprise Edition Oracle Database 12c R1 Standard Edition Oracle Database 12c R1 Standard Edition One	Oracle 12c, Oracle
Microsoft SQL Server 2005 Enterprise Edition Microsoft SQL Server 2005 Standard Edition	SQL Server 2005, SQL Server
Microsoft SQL Server 2008 Enterprise Edition Microsoft SQL Server 2008 Standard Edition	SQL Server 2008, SQL Server
Microsoft SQL Server 2012 Enterprise Edition Microsoft SQL Server 2012 Standard Edition	SQL Server 2012, SQL Server
Symfoware V11 Enterprise Edition Symfoware V11 Standard Edition Symfoware V12 Standard Edition	Symfoware Server, Symfoware

Name	Abbreviation
PostgreSQL 9.1	PostgreSQL

Export Regulations

Fujitsu documents may include special technology regulated by foreign exchange and foreign trade regulation laws. If special technology is included, permission must be obtained under these laws before exporting a document or providing it to a non-resident.

Trademarks

- Microsoft, Windows, and Windows Server are trademarks or registered trademarks of Microsoft Corporation in the United States and other countries.

- Oracle and Java are registered trademarks of Oracle and/or affiliates. Other names may be trademarks of their respective owners.

- Linux is a registered trademark or trademark of Linus Torvalds in the United States and other countries.

- Red Hat is a trademark or registered trademark of Red Hat Inc. in the United States and other countries.

- SQL Server is a Microsoft Corporation (USA) product.

- All other company and product names referred to in this document are general trademarks or registered trademarks of their respective owners.

Requests

- This document may be revised without notice.

- Do not pass on this document to other parties without permission.

- Our company bears no responsibility for third-party infringements of patent or other rights arising out of use of the data contained in this document.

Copyright

Contents

Chapter 1   Overview of Analytics

The Interstage Business Process Manager Analytics (BPMA) software provides the following functionality:

- Business Activity Monitoring (BAM), including data collection and analysis

- Process visualization feature of Process Discovery

Use it with the Interstage Business Process Manager (BPM) for effective business process visualization and enhance your business performance.

1.1   Business Activity Monitoring - BAM

A comprehensive Business Activity Monitoring (BAM) suite enables all your business processes to be monitored using one centralized application. The BAM suite is designed to complement and effortlessly integrate with the Interstage BPM, and also, access and utilize other data sources in a wider business environment.

Analytics facilitates superior data collection, analysis, and display functionality associated with the modern BAM architecture. All this can be scheduled to occur real-time or periodically.

Moreover, the results of an analysis can be viewed graphically using the Analytics Dashboard or other actions could be initiated, such as e-mails to responsible parties or invoke another application. Data could also be exported to another analysis system, for instance, as a .csv log file.

1.2   Process Visualization (Process Discovery)

Process Discovery enables you to assess the level to which an IT system suits a particular business. It allows you to analyze the data accumulated in an IT system to yield critical information about how the IT system is used real time.

The data accumulated in the database tables of an IT system is fed into Process Discovery and a chart showing the business processes is generated as the output. These charts can be in many different styles and the analysis can be from different viewpoints, making it easier to identify issues and problems of an IT system or business process.

The data fed into Process Discovery is in the text, Comma-Separated Value (CSV), format. It is a simple matter of converting the data, stored in a database or worksheets, into a CSV, and then, feeding it into Process Discovery. So, if you already have the CSV data, you can easily analyze your business processes using Process Discovery.

Chapter 2   What Analytics Does

The Interstage Business Process Manager Analytics (BPMA) is capable of the following three types of analysis:

- Analysis of real-time information/data from existing systems such as applications, databases, and ERP systems

- Analysis of the performance of business processes as an Interstage BPM extension

- "As-Is" analysis of business processes based on results

2.1   Components of Analytics

The following graphic demonstrates the role of Analytics in a business environment that uses the Interstage Business Process Manager (BPM) as well as other application/ERP components.
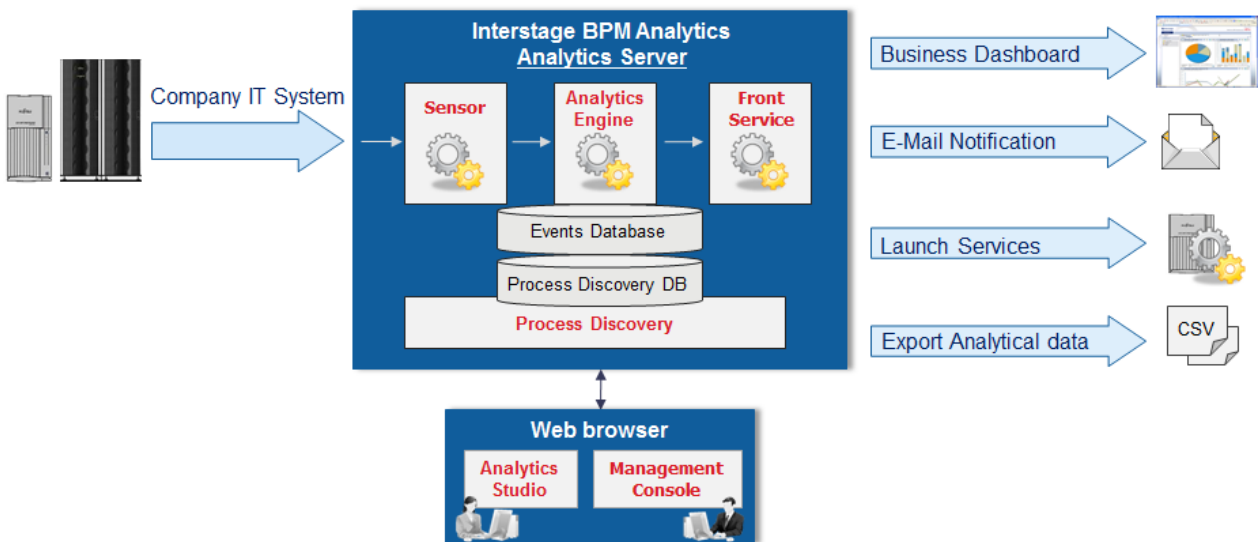
Analytics comprises five main modules:

Sensors

Enable connection with other business systems and gather business data, referred to as events, from those systems; events are generated from text or .csv files, databases, and Interstage BPM systems

Analytics Engine

Captures all data, ensures it is in the common format, and then, analyzes it according to the business rules set up in the Analytics Studio

Front Service

Represents the event information in a dashboard-style display; data can be either presented through e-mail notifications or actions, such as calling another application or event, or simply, exported as raw data to another application

Process Discovery

Process Discovery provides two features: Process Generator and Process Discovery. Process Generator generates business processes from CSV-format business logs. Process Analyzer analyzes the business processes generated by Process Generator and produces visualizations of these processes. If the Interstage BPM business process data is collected as Analytics events, this data is shown as processes.

Analytics Studio

Web-based tool for configuring the settings of data collection, data analysis, and presentation

Management Console

Web-based administration tool for operating Analytics Server start/stop, and monitoring the data status and the Events databases

2.2　How Analytics Works

This chapter explains how the following components work.

- Sensors

- Analytics Engine

- Front Service

- Process Discovery

2.2.1　Sensors

Sensors collect business data from the entire business and store it as events in the Events database. The scope of Analytics collection capabilities extends beyond the processes defined in the Interstage BPM and can include data and information from other data sources that are within the business.

Events to be collected and the configuration of their sensors is defined and configured in the Analytics Studio.

Analytics can monitor the following systems.

Workflow of Interstage BPM processes

Analytics can collect, as events, a variety of historical information such as process start, activity start and completion, user-defined information, as well as performance information on how well the processes within the IBPM system are running.

RDB

At regular intervals, the Analytics RDB monitoring function can be set to collect data from business databases and store them in the Events database in a common format.

Text files

Data that has been saved in text files during business system operations can also be collected as events. Analytics can collect CSV and fixed-recorded file, which is usually used by legacy systems.

Text file monitoring can be used to periodically browse within files and collect, as events, the parts that have been added.

Additionally, collection conditions can be set so that only specific data that matches those conditions is collected. For example, an event "Monitoring of travel-expense applications that exceed a total of $100 at 30 minute intervals" can be implemented using Analytics.

Schedules and rules for event collection can also be set.

📔 Note

There are two types of text file sensors, including sensors that operate as web applications and sensors that operate as Microsoft Windows Services (Windows Service Edition Text File Sensor). The Windows Service Edition Text File Sensor can be used to collect monitoring data from client systems.

Setting event collection schedules

Users can set schedule for event collection, such as, specific time intervals, every week, or at the end of each month. These settings are configured using the Analytics Studio.

2.2.2   Analytics Engine

The ability to calculate statistical data and run functions on events to produce meaningful, custom-made analysis is the most important Analytics feature.

These calculations are done by the Analytics Engine that uses data stored in the Events database.

The event processing function performs statistical calculation and alert assessment on the collected events.

Statistical calculation

Using the Analytics Studio, statistical settings or rules can be defined for events. For example, the sum or average of events can be calculated. Statistical results are managed as statistical events in the Events database.

Alert assessment

The Analytics Engine assesses the collected events by setting conditions that use threshold values. If these thresholds are exceeded, an alert is triggered. Alerts can, in turn, become events of statistical results, saved in the Events database.

Examples of alerts include:

  - Checking events by parameter values

    Example: When monthly sales are less than $5,000,000

  - Checking the progress of a process

    Example: When a process step is not completed three days before the next step's completion due date

2.2.3   Front Service

When the Analytics Server starts working, it collects events from variety of sources. These are stored in the Events database using a common format. There the data is processed by the Analytics Engine and the results are also stored in the Events database, available for presentation.

Presentation can occur in a number of ways.

While the most obvious use of the results is to present the information in a dashboard so that it is visible and useful to managers and other analysts, the results can also be used to trigger a new event, for example, to send an e-mail or invoke another application.

The analysis data can also be output as a file to be used in another analysis system.

Dashboard

The status of business activities can be displayed using graphs or tables, while the status of alerts can be displayed real time. This enables rapid and effective response to events (for example, errors) that can occur in a business process.

The Analytics Dashboard is defined as a profile in the Analytics Studio. Components such as charts, alerts, and statistics can be included in each profile.

Alerts and escalation

In addition to presenting data and results on the Analytics Dashboard, various other operations are possible with the Analytics Front Service.

The alerts, created using the Analytics Studio Alert wizard, are used to create alert tabs in the Analytics Dashboard (as described in the previous section), and then, the following operations are possible.

Use alerts as trigger mechanisms to launch other events or send messages.

Alerts / escalation

    The content of an alert can be sent to a user by e-mail.

    Set this in the Analytics Studio.

Calling external applications

    Escalation can also involve a call to external services or applications. The launch can be set as the response to an event.

Set this in the Analytics Studio.

Linking data with external tools

Data can also be exported to an environment outside Analytics.

The monitoring results in the Events database can be output to CSV files, so that this data can be linked to other systems or analyzed using tools, such as Microsoft Excel. Use the management command to output data in the CSV format.

Reports

Use the Analytics Studio **Reports** tool to generate reports.

Reports have their own schedule and profile.

Reports can also include one or more KPI charts or alerts list.

When you define reports in the Analytics Studio, set the schedule for creating reports and specify the Profile layouts to be used for generating them.

2.2.4 Process Generation using Process Generator

Processes are generated by the Process Discovery - Process Generator using the data accumulated in a business database. Key features of process generation include:

- Profiling

  Analyzes the peculiarities of the data and extracts candidates that can be used as time information and event types

- Correlation Candidate Selection

  Allows you to select the correct event types and set the correlations between events

- Process Data Generation

  Takes the information determined above and creates processes from the CSV data, storing the results in the Process Discovery database

2.2.5 Process Analysis using Process Analyzer

The Process Discovery - Process Analyzer analyzes and displays the business process data generated by the Process Generator or Interstage BPM. Key features of process analysis include:

- Process Analysis

  Allows you to filter process data, display processes, and detect bottlenecks

- Route Analysis

  Analyzes routes or order of events in processes; uses the principles that if the order of occurrence of events is the same in processes, the same route is followed

- Exceptional Route Analysis

  Analyzes processes that do not have the same route or order of events as majority of processes in a business environment

- Parallel Analysis (only when displaying Interstage BPM)

  Allows you to view the extent to which simultaneous events overlap

Chapter 3   Setting Up and Configuring Analytics

This chapter provides an overview of how to setup and configure the Interstage Business Process Manager Analytics (BPMA).

Setting Up BAM

The following illustration shows the setup procedure.
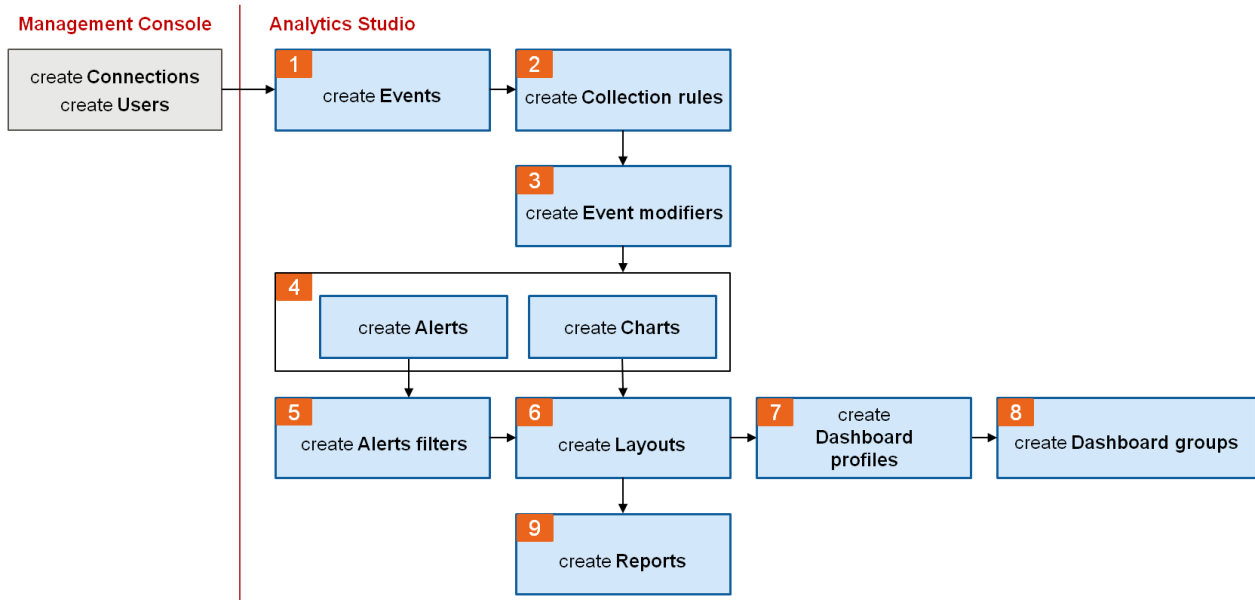
Setting Up Process Discovery

The following illustration shows the setup procedure.
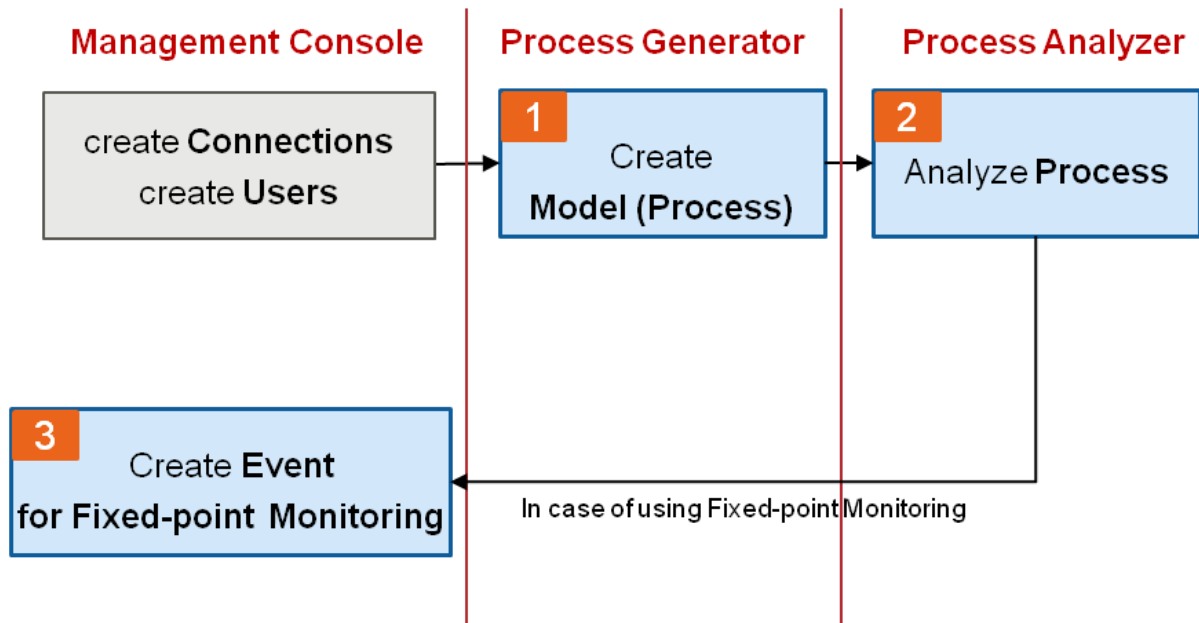
Chapter 4 Overview of BAM functionalities and Some Examples

This chapter gives an overview of BAM functionalities and some examples.

4.1 Overview of Event Collection

4.1.1 Data Types and Monitoring Methods

Configure event collection to match business requirements by combining the settings for data types and monitoring methods.

The following table describes the two data types.

Table 4.1 Data types

Data types	Description
Master Data Management	Covers data used for updating, such as inventories and summaries.
Transaction Data Management	Covers data that is added, such as sales and transaction records.

The following three monitoring methods are available:

- Add difference records

- Easy mirroring

- Full history

The behavior of event collection depends on the combination of the data and monitoring methods used, as described in the following table.

Table 4.2 Differences in behavior depending on the combination of data and monitoring methods used

Data types	Monitoring method	Event collection behavior
Master Data Management	Add difference records	Only records that have been added to a business database are added to Analytics data as new records.
Master Data Management	Easy mirroring	Records added to a business database are added to Analytics data as new records. If records are updated on the business database, they are similarly updated in the Analytics data.
Master Data Management	Full history	All data is gathered with every collection and stored in Analytics.
Transaction Data Management	Add difference records	* Behavior is the same as the Master Data Management/Add difference records combination.
Transaction Data Management	Easy mirroring	* Behavior is the same as the Master Data Management/ Easy mirroring combination.
Transaction Data Management	Full history	Records added to a business database and data updated on a business database are both added to Analytics data as new records.

Master Data Management with Add Difference Records

When event collection occurs once a day, and where the following changes are apparent in the data between one day and the next, only data that is added is reflected as updates in the Events database. The behavior will be the same for Transaction Data Management with Add difference records.
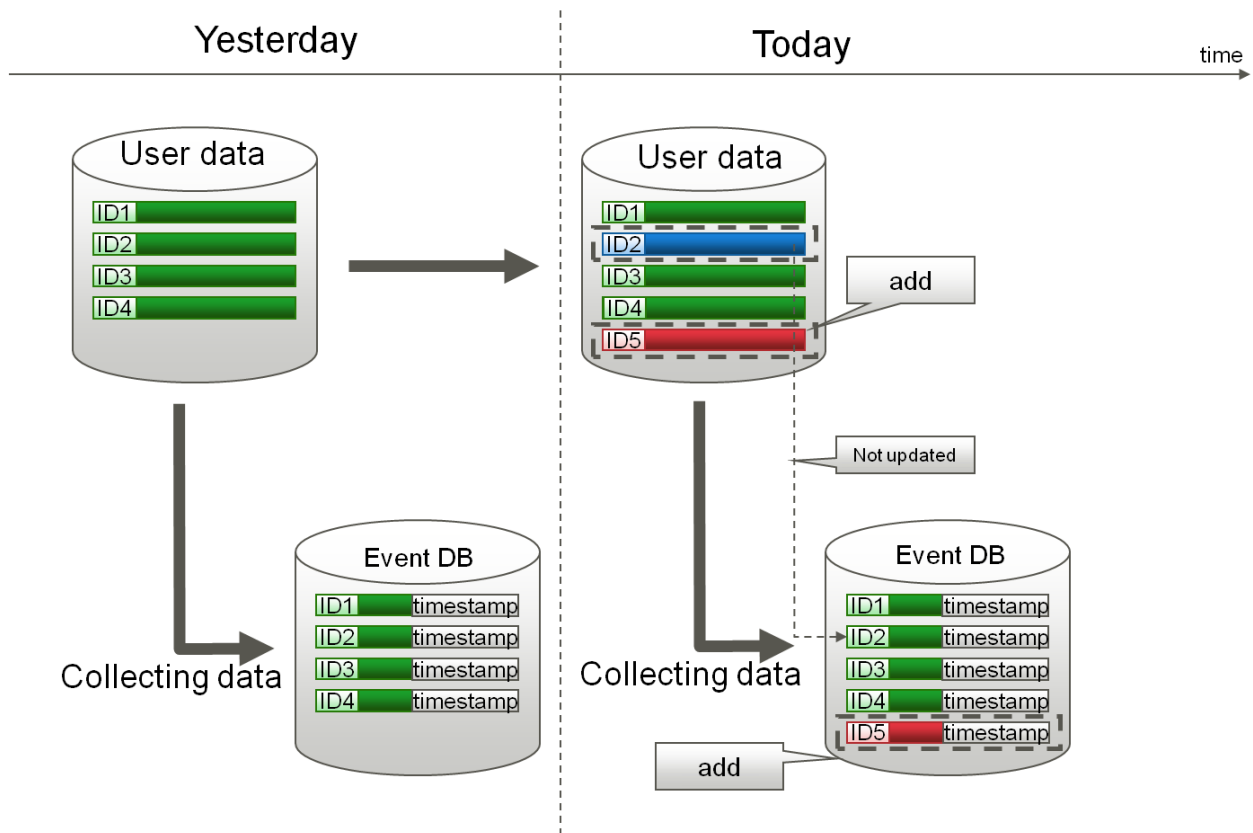
Master Data Management with Easy Mirroring

When event collection occurs once a day, and where the following changes are apparent in the data between one day and the next, data that is added is reflected as a new event in the Events database, and updated data is reflected in the Events database as updates. The behavior will be the same for Transaction Data Management with Easy mirroring.
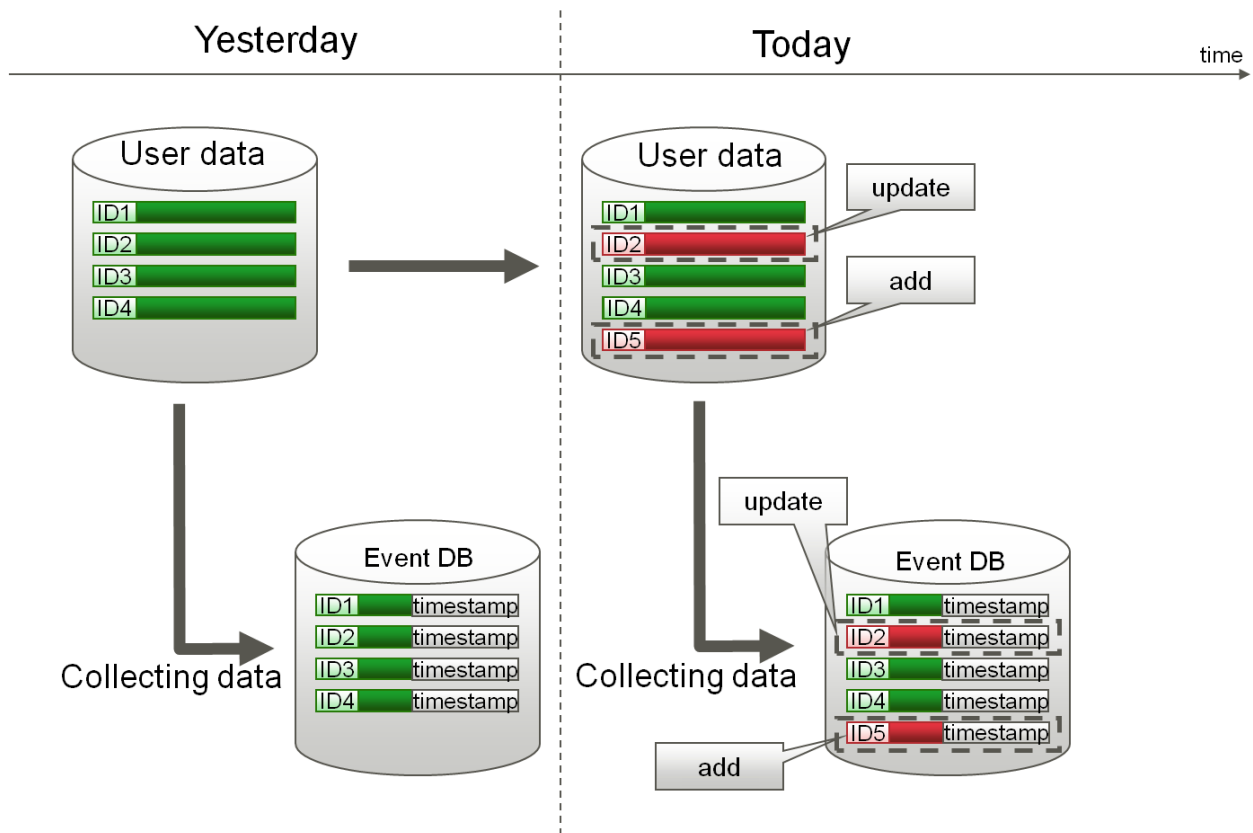
Master Data Management with Full History Monitoring

When event collection occurs once a day and when the following changes are apparent in the data between one day and the next, all monitored data is collected and stored in the Events database.
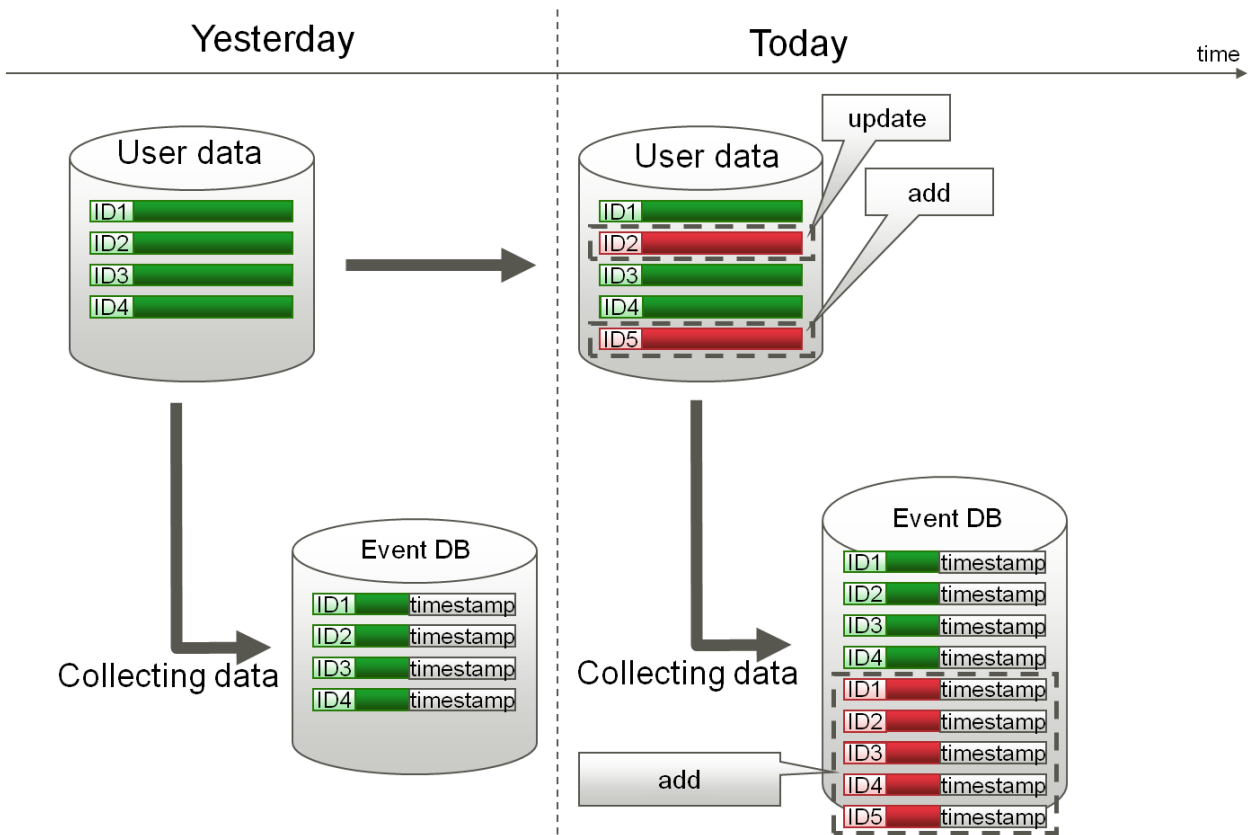
Transaction Data with Full History Monitoring

When event collection occurs once a day, and where the following changes are apparent in the data between one day and the next, data that is updated and data that is added is stored in the Events database as new events.
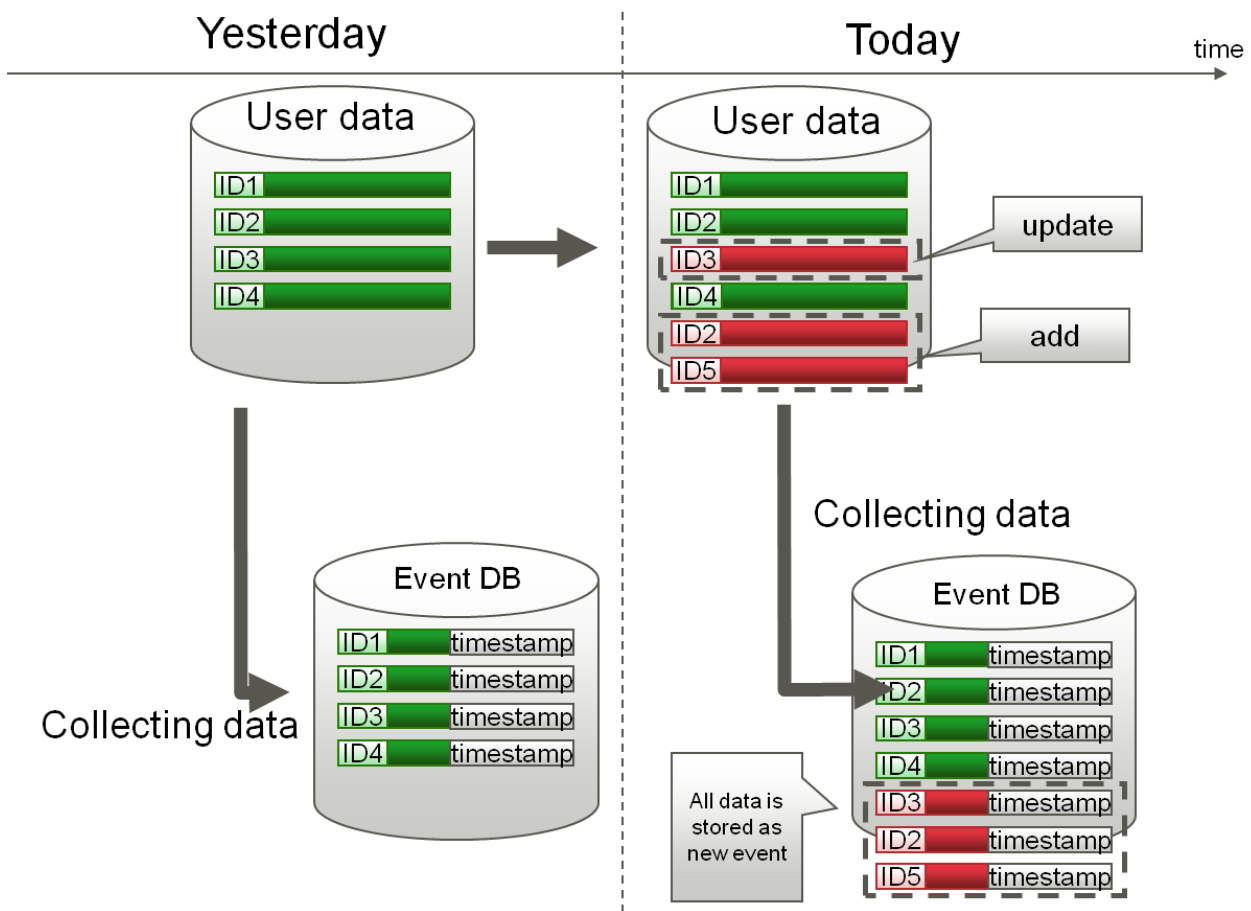
4.2 Event Collection Example

4.2.1 RDB Sensor (To Detect Added Records)

In the following example, an RDB sensor collects business data from a business database and stores it in the Events database as a general event. In this example, the business table contains a time column that records the time when events occur; this time can be used to distinguish which records should be collected as events. The RDB sensor only collects new records from the business database and adds them to the Events database.
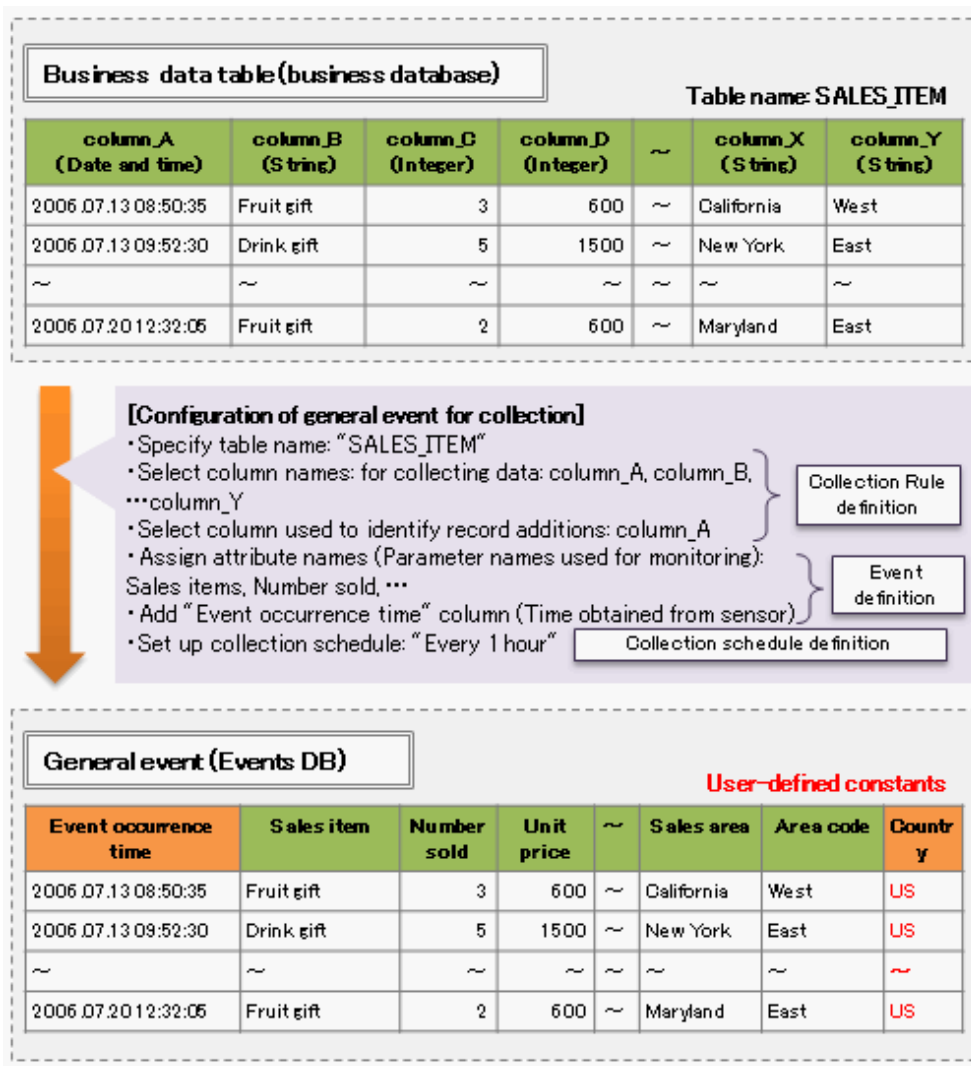
Specify which table and column to collect data from in a collection task. The example shows that data is to be collected from the SALES_ITEM table, starting at column_A and ending at column_Y. The column used to distinguish events is set as column_A (the column used to record the time when events occurred). A user-defined constant has also been added (the "Country" column with the value "Japan").

Next, use an event to specify the item names and data types for the items that store the collected data. The example shows items that store data from "column_B" have been labeled "Sales item", and the items that store data from "column_C" have been labeled "Number sold".

Finally, use a collection schedule to specify the collection timing. In this example, the collection schedule has been set to "every hour". The collection task will be executed every hour, so long as the sensor is running.

4.2.2 RDB Sensor (To Detect Updated Records)

In the following example, an RDB sensor collects business data from a business database and stores it in the Events database as a general event. In this example, the business table does not contain a time column that records the time when events occur, so time cannot be used to distinguish which records should be collected as events. All records from the business database are collected and added to the Events database. This information can be used to manage stock level trends, for example.

Business data table (business database)

Table name: STOCK_ITEM

column_A(String)	column_B(Integer)	~	column_X(String)	column_Y(String)
Fruit gift	2100	~	California	West
Drink gift	2800	~	New York	East
Flower gift	4550	~	Maryland	East

[Configuration of general event for collection]
· Specify table name: "STOCK_ITEM"
· Select column names for collecting data: column_A, column_B, ···column_Y
· Assign attribute names (Parameter names used for monitoring)
· Add "Event occurrence time" column (Time obtained from sensor)
· Set up collection schedule: "Every 1 hour"

Collection rule definition

Event definition

Collection schedule definition

General event (Events DB)

Event occurrence time	Inventory Item	Number of stocks	~	Place of inventory	Area code F
2006.07.13 08:00:00	Fruit gift	2500	~	California	West
2006.07.13 08:00:00	Drink gift	3000	~	New York	East
2006.07.13 08:00:00	Flower gift	4700	~	Maryland	East
2006.07.13 09:00:00	Fruit gift	2100	~	California	West
2006.07.13 09:00:00	Drink gift	2800	~	New York	East
2006.07.13 09:00:00	Flower gift	4550	~	Maryland	East

Records collected and added at 8:00

Records collected and added at 9:00

All three records ("Fruit gift", "Drink gift" and "Flower gift") are collected every hour and added to the Events database. This makes it possible to work out, for example, that the stock levels for "Fruit gift" have dropped from 2500 at eight o'clock to 2100 at nine o'clock.

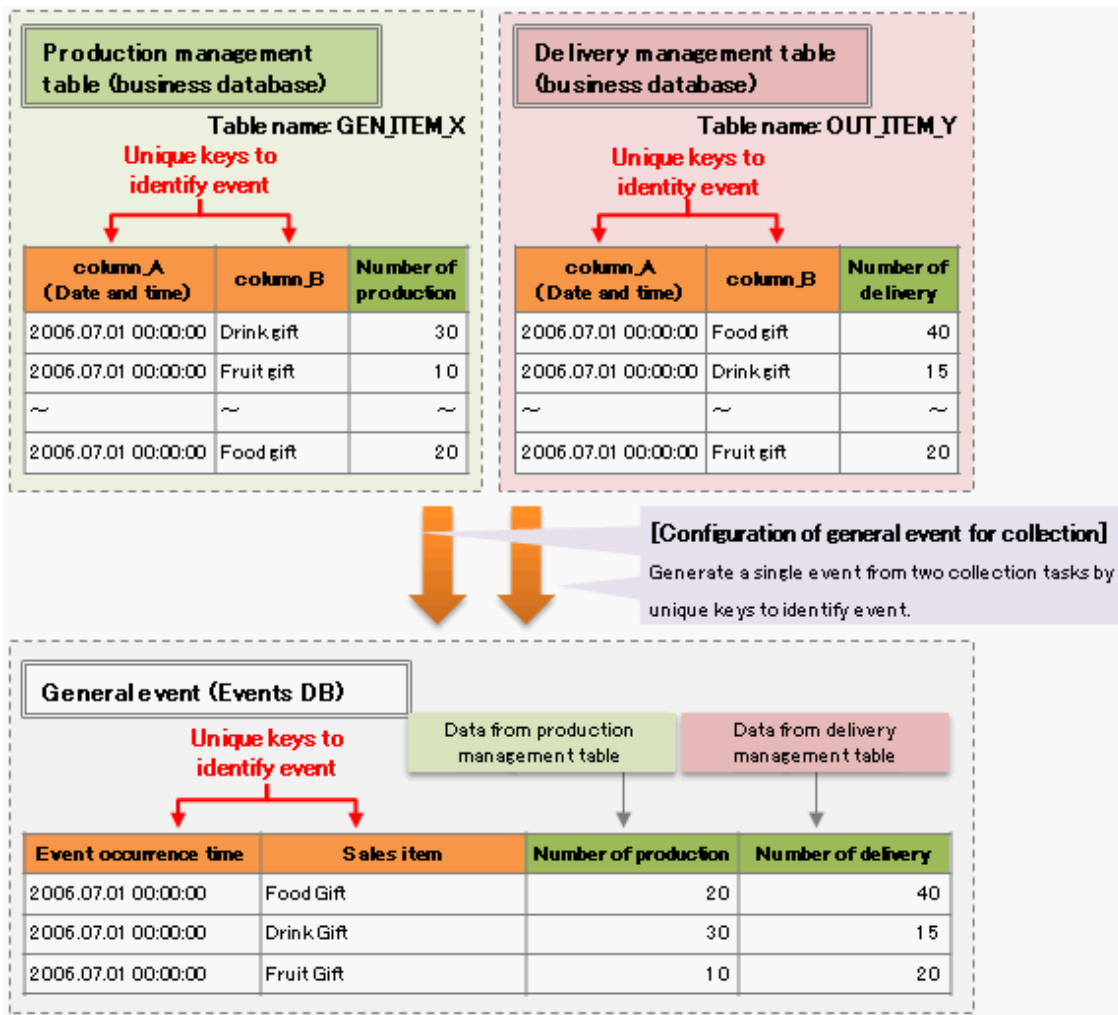## 4.2.3   RDB Sensor (Merging from Separate Tables)

In the following example, an RDB sensor collects data from two business tables in a business database and stores this data in the Events database as a single general event.

Business data table X (Business database)
Table name: STOCK_ITEM_X

column_A(Date and time)	column_B	column_C	column_D
2006.07.13 08:50:35	Drink gift	30	New York
2006.07.13 08:52:30	Fruit gift	10	Florida
~	~	~	~
2006.07.20 10:32:05	Drink gift	20	Maryland

Business data table Y (Business database)
Table name: STOCK_ITEM_Y

column_A(Date and time)	column_B	~	column_L	column_M
2006.07.11 14:07:35	Nevada	~	Food gift	40
2006.07.13 18:20:30	California	~	Drink gift	15
~	~	~	~	~
2006.07.20 09:32:05	Oregon	~	Fruit gift	20

[Configuration of general event for collection]
Two collection tasks allow to manage as a single event

General event (Events DB)

User-defined constants

Event occurrence time	Sales item	Number sold	Sales area	Area code	
2006.07.11 14:07:35	Food gift	40	Nevada	West	Data from business data table Y
2006.07.13 08:50:35	Drink gift	30	New York	East	Data from business data table X
2006.07.13 08:52:30	Fruit gift	10	Florida	East	Data from business data table X
2006.07.13 18:20:30	Drink gift	15	California	West	Data from business data table Y
~	~	~	~	~	
2006.07.20 09:32:05	Fruit gift	20	Oregon	West	Data from business data table Y
2006.07.20 10:32:05	Drink gift	20	Maryland	East	Data from business data table X

Create a collection task for business table X, and then create an event to store the collected data. Next, create a collection task for business table Y, and then specify how business table Y and the event created with the collection task for business table X correspond.

User-defined constants can also be added as identifiers used when the data is merged. In this example, an "Area code" has been added.

4.2.4 RDB Sensor (Merging into the Same Event from Separate Tables)

In the following example, data is collected from two tables in a business database by an RDB sensor, merged into one event, and stored in the Events database.

Create collection tasks for both the Production table and the Shipping table, mapping the attributes that represent items sold and dates to the event keys defined in the general event. When events are generated based on the data from two tables, as shown above, the event that arrives first is added as a new event. The event arriving later then uses the event keys to determine the relevant attributes and updates them.

4.2.5   RDB Sensors (Comparing against the Master Table)

In the following example, an RDB sensor collects and merges data from two business tables in a business database and stores this data in the Events database as a single general event.

Sales order table (business database)

Table name: SALES_TRAN

column_A (Date and time)	column_B	column_C	column_D
2006.07.13 08:50:35	ITEM0002	30	New York
2006.07.13 08:52:30	ITEM0001	10	Maryland
2006.07.15 10:08:00	ITEM0002	10	Maryland
~	~	~	~
2006.07.20 10:32:05	ITEM0003	20	Florida

Item master table (business database)

Table name: ITEM_NAME

column_A	column_B
ITEM0001	Fruit gift
ITEM0002	Drink gift
ITEM0003	Food gift
~	~

[Configuration of general event for collection from sales order table]
· Specify table name (FROM clause): SALES_TRAN, ITEM_NAME
· Join key (WHERE clause): SALES_TRAN.column_B=ITEM_NAME.column_A }Collection rule definition
· Select column names to collect data: ITEM_NAME.column_B, SALES_TRAN.column_C, ···
· Assign attribute names (Parameter names used for monitoring)
· "Event occurrence time" column: SALES_TRAN.column_A }Event definition

[Configuration of general event for collection from item master table]
· Specify table name (FROM clause): SALES_TRAN INNER JOIN ITEM_NAME ON SALES_TRANS.column_B=ITEM_NAME.column_A
· Join key (WHERE clause): }Collection rule definition
· Select column names to collect data: ITEM_NAME.column_B, SALES_TRAN.column_C, ···
· Assign attribute names (Parameter names used for monitoring)
· "Event occurrence time" column: SALES_TRAN.column_A }Event definition

General event (Events DB)

Event occurrence time	Sales item	Number sold	Sales area
2006.07.13 08:50:35	Drink gift	30	New York
2006.07.13 08:52:30	Fruit gift	10	Maryland
2006.07.15 10:08:00	Drink gift	10	Maryland
~	~	~	~
2006.07.20 10:32:05	Food gift	20	Florida

Example 1: Using join keys (collecting general events 1)

A collection task is created using "column_B" in the Sales slip table and "column_A" in the Item master table as join keys. This makes it possible to collect business data from multiple tables and integrate this data as a single event. In this example, "ITEM0001" in "column_B" is associated with the "Fruit gift" sales item in the Item master table and stored in the Events database. Doing so makes monitoring windows and CSV file output easier to understand.

### Example 2: Using a FROM clause (collecting general events 2)

There is no join key in example 2, so items are associated with the Item master table by using an "INNER JOIN" in the FROM clause. The condition in "column_B" of the Sales slip table should be the same as "column_A" of the Item master table, which is defined using the table name specification for the collection task.
The results are the same as example 1.

4.2.6 Text File Sensors (Fixed File Names)

In the following example, text file sensors collect data from a file with a fixed name and store this data in the Events database as general events. This can be used with Business Processes where logs are always output with the same file names.
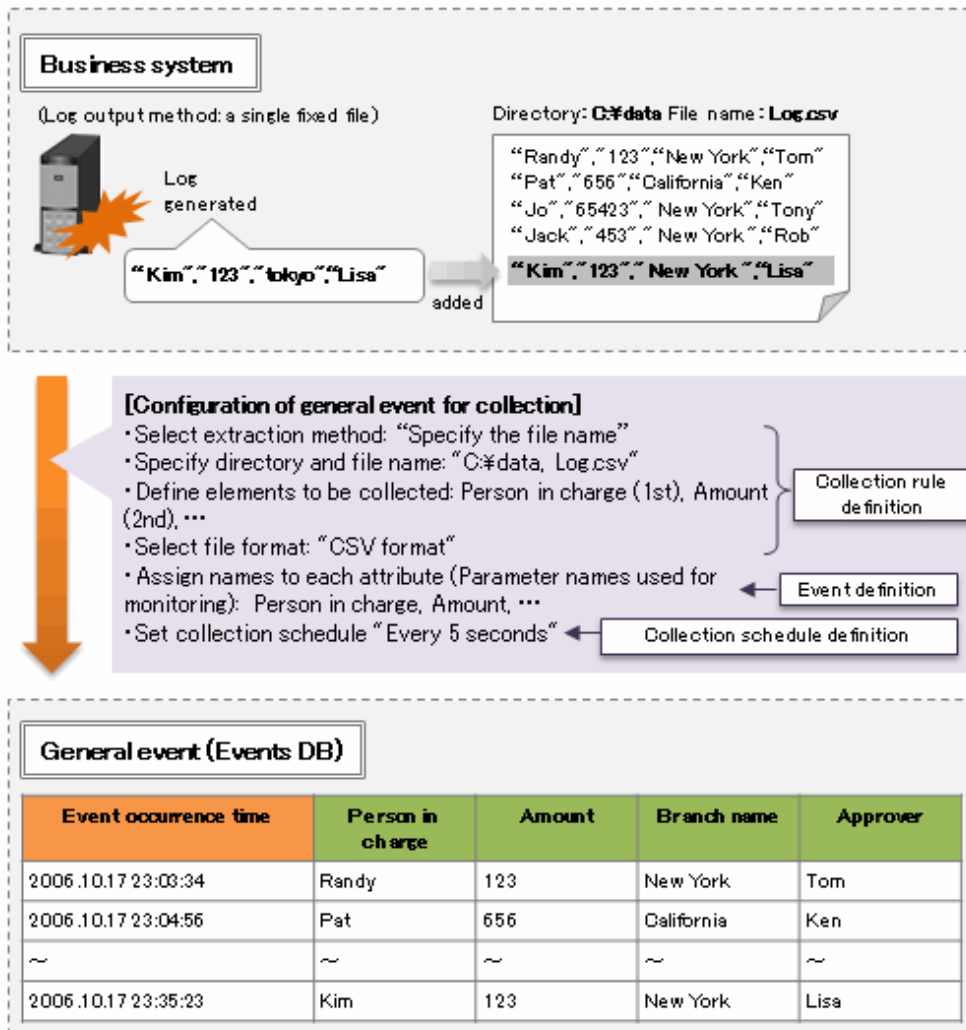
In the collection task, specify the file and location in the file of the data to be collected. In this example, the first, second, third and fourth data elements in the file "Log.csv" have been specified.
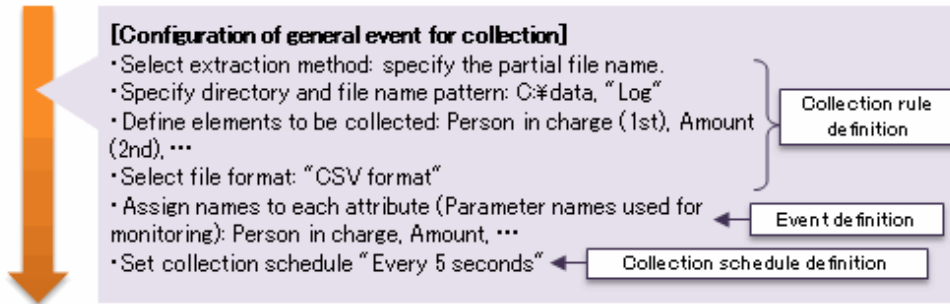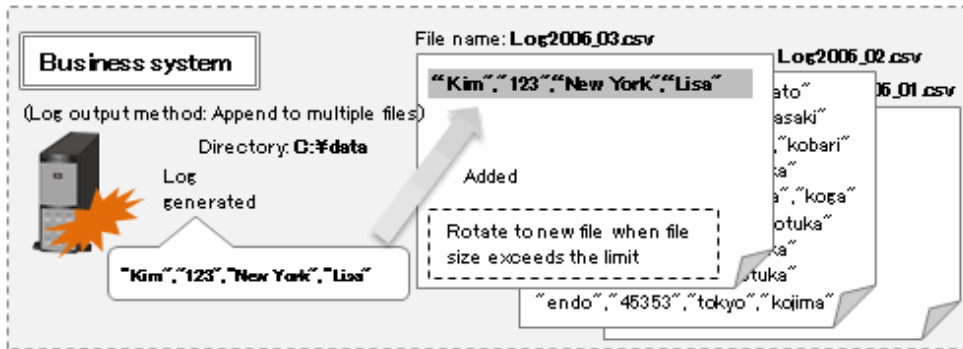
Next, use an event to specify the item names and data types for the items that store the collected data. In this example the labels of the four items are "Person in charge", "Amount", "Branch name", and "Approver".

Finally, use a collection schedule to specify the collection timing. In this example, the collection schedule has been set to "every 5 seconds". The collection task will be executed every 5 seconds so long as the sensor is running.

It is also possible to specify that all data within the file is to be collected with each collection.

4.2.7 Text File Sensors (The partial file name)

In the following example, text file sensors collect data from files whose names contain a particular string, and store this data in the Events database as general events. This can be used with Business Processes where new log files are created once the current log file reaches a certain size and where these log file names contain a particular string.

Business system

(Log output method: Append to multiple files)

Directory: C:¥data

Log generated

File name: **Log2006_03.csv**

"Kim","123","New York","Lisa"

Added

Rotate to new file when file size exceeds the limit

"Kim","123","New York","Lisa"

Log2006_02.csv

...ato"
...asaki"
..."kobari"
...a","koga"
...otuka"
...a"
...tuka"
"endo","45353","tokyo","kojima"

...6_01.csv

[Configuration of general event for collection]
・Select extraction method: specify the partial file name.
・Specify directory and file name pattern: C:¥data, "Log"
・Define elements to be collected: Person in charge (1st), Amount (2nd), ···
・Select file format: "CSV format"
・Assign names to each attribute (Parameter names used for monitoring): Person in charge, Amount, ···
・Set collection schedule "Every 5 seconds"

Collection rule definition

Event definition

Collection schedule definition

General event (Events DB)

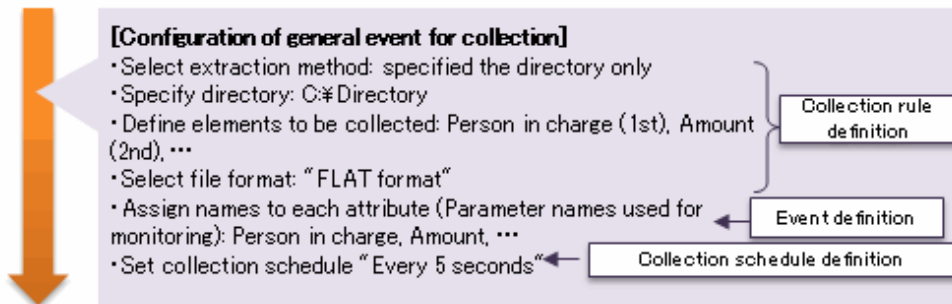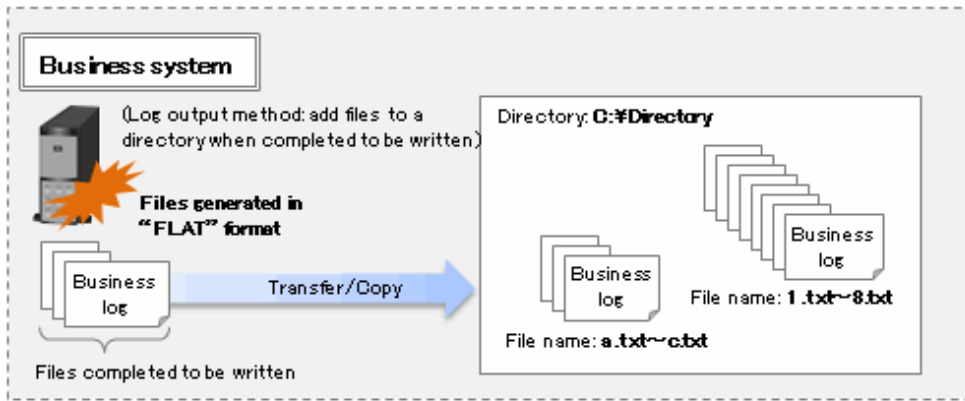| Event occurrence time	Person in charge	Amount	Branch name	Approver
2006.10.17 23:03:34	Randy	123	New York	Tom
2006.10.17 23:04:56	Pat	656	California	Ken
~	~	~	~	~
2006.10.17 23:35:23	Kim	123	New York	Lisa

In the collection task, specify the file and location in the file of the data to be collected. In this example, the first, second, third and fourth data elements in a file that includes the string "Log" have been specified.

Next, use an event to specify the item names and data types for the items that store the collected data. In this example the labels of the four items are "Person in charge", "Amount", "Branch name", and "Approver".

Finally, use a collection schedule to specify the collection timing. In this example, the collection schedule has been set to "every 5 seconds". The collection task will be executed every 5 seconds so long as the sensor is running.

4.2.8   Text File Sensors (Transfer File Group)

In the following example, text file sensors collect data from all of the files in a particular directory and store this data in the Events database as general events. This can be used with Business Processes where log information is transferred or copied to a particular directory as a file, once it has been written completely.

In the collection task, specify the file and location in the file of the data to be collected. In this example, the first, second, third and fourth data elements in all the files in the folder "C:\Directory" have been specified.

Next, use an event to specify the item names and data types for the items that store the collected data. In this example the labels of the four items are "Person in charge", "Amount", "Branch name", and "Approver".

Finally, use a collection schedule to specify the collection timing. In this example, the collection schedule has been set to "every 5 seconds". The collection task will be executed every 5 seconds so long as the sensor is running.

4.3 Event Processing

4.3.1 Event Calculation (Calculation Operations)

In the following example, new calculation events are output by performing calculations on the general events stored in the Events database.

[Example 1: configuration of statistical event]
・Calculation period: 1 hour
・Output item 1: calculate "Total sales" from "Sales" (sum)
・Output item 2: use the same "Area code"
・Group attribute: "Area code"

General event (Events DB)

Event occurrence time	Item name	Sales	~	Sales area	Area code
~					
2006.07.13 07:50:00	Fruit gift	1800	~	New York	EAST
2006.07.13 08:20:00	Drink gift	1000	~	California	WEST
2006.07.13 08:45:00	Fruit gift	2100	~	Nevada	WEST
2006.07.13 09:10:00	Fruit gift	5300	~	California	WEST
2006.07.13 09:10:00	Drink gift	2800	~	Nevada	WEST
2006.07.13 09:15:00	Fruit gift	1600	~	California	WEST
2006.07.13 09:50:00	Drink gift	1800	~	New York	EAST
2006.07.13 10:00:00	Drink gift	2900	~	Nevada	WEST
2006.07.13 10:05:00	Fruit gift	4500	~	New York	EAST
2006.07.13 11:30:00	Drink gift	600	~	Nevada	WEST
~					

Statistical event (Events DB)

Event occurrence time	Total sales	Area code
~		
2006.07.13 09:00:00	0	EAST
2006.07.13 09:00:00	3100	WEST
2006.07.13 10:00:00	1800	EAST
2006.07.13 10:00:00	9700	WEST
2006.07.13 11:00:00	4500	EAST
2006.07.13 11:00:00	2900	WEST
~		

Event occurrence time	Total sales	Average sales
~		
2006.07.13 09:00:00	3100	1550
2006.07.13 10:00:00	11500	2875
2006.07.13 11:00:00	7400	3800
~		

[Example 2: configuration of statistical event]
・Calculation period: 1 hour
・Output item 1: calculate "Total sales" from "Sales" (sum)
・Output item 2: calculate "Sales average" from "Sales" (average)
・Group attribute: Not specified

Example 1: Calculating sales totals grouped by area code

Calculations are performed on general events every hour. A group attribute has been specified, and so separate calculations are made for each group of events with the same group attribute (in this case, "area code"). For example, at eight o'clock calculations are made for the general events that have occurred in the last hour (since seven o'clock). Since there are no general events with the area code "KANTO", the "Total sales" for this group is "0". There are two events with the area code "KANSAI" (1000 for the "First tea of the season" event with the "Osaka" sales area code, and 2100 for the "Strawberries" event with the "Kyoto" sales area code) and so the "Total sales" is "3100".

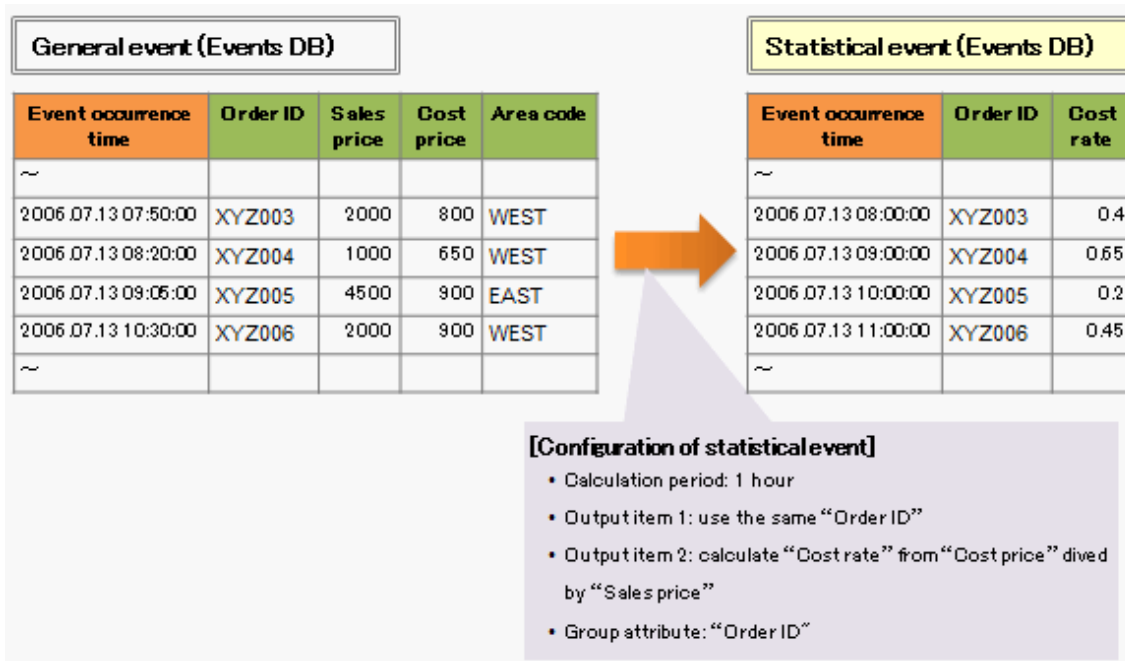## Example 2: Working out average values from sales totals

Calculations are performed on general events every hour. Group attributes have not been specified, and so one statistics event is output for each calculation. For example, at eight o'clock, the "Total sales" and "Sales average" are calculated for the following general events that have occurred in the hour since seven o'clock.

- 1000 for the "First tea of the season" event with the "Osaka" sales area code

- 2100 for the "Strawberries" event with the "Kyoto" sales area code

The "Total sales" is 3100 and the "Sales average" is 1550.

4.3.2 Event Calculation (Arithmetic Calculations)

In the following example, new statistics events are output by performing arithmetical operations on the general events stored in the Events database.

General event (Events DB)

Event occurrence time	Order ID	Sales price	Cost price	Area code
~				
2006.07.13 07:50:00	XYZ003	2000	800	WEST
2006.07.13 08:20:00	XYZ004	1000	650	WEST
2006.07.13 09:05:00	XYZ005	4500	900	EAST
2006.07.13 10:30:00	XYZ006	2000	900	WEST
~				

Statistical event (Events DB)

Event occurrence time	Order ID	Cost rate
~		
2006.07.13 08:00:00	XYZ003	0.4
2006.07.13 09:00:00	XYZ004	0.65
2006.07.13 10:00:00	XYZ005	0.2
2006.07.13 11:00:00	XYZ006	0.45
~		

[Configuration of statistical event]
- Calculation period: 1 hour
- Output item 1: use the same "Order ID"
- Output item 2: calculate "Cost rate" from "Cost price" dived by "Sales price"
- Group attribute: "Order ID"

This example generates the cost price ratio from the sales and cost price information in the sales slip.

Refer to "Editing Event Modifiers" in the *Studio Guide* for information on how to set the cost:price ratio using the Analytics Studio.

4.3.3 Evaluating Alerts (Parameter Evaluation)

In the following example, alert messages are output by setting up alert rules for evaluating the parameters of single events.

General event (Events DB)

	Event occurrence time	Sales item	Number sold	Unit price	Sales	~	Sales area	Area code
a	2006.07.13 08:50:35	Fruit gift	3	600	1800	~	New York	EAST
b	2006.07.13 08:52:30	Drink gift	5	1500	7500	~	Nevada	WEST
	~	~	~	~	~	~	~	~
c	2006.07.20 10:32:05	Fruit gift	2	600	1200	~	California	EAST

[Configuration of alert rule]

- Thresholds and character strings can be used for comparison.
- "and" and "or" operations can be used to create complex conditions.
- The results of arithmetic operations between columns can also be used in conditions.

Example of alerting conditions		Matched events	Example of alerting messages
"Sales item" is equal to "Fruit gift".	→	a, c	Fruit gift was ordered.
"Number sold" is 5 or greater.	→	b	Order of 5 or greater was placed.
"Number sold" times "unit price" is 1500 or less.	→	a	Order of 1500 yen or less was placed.
"Area code" is equal to "KANSAI" and "Number sold" is "5" or greater.	→	c	Order for 5 or more items was placed in KANSAI.
"Sales item" includes "Fruit gift" and "Sales area" is "Osaka".	→	c	Order related to Fruit gift was placed in Osaka.
"Number sold" is "10" or greater, or "Sales" is "5000" or more.	→	b	Large order was placed.

For example, the alert rule '"Sales item name" is equal to "Strawberries"' will match events (a) and (c), and the message "An order for strawberries was placed" will be displayed in the **Alert View** of the Dashboard.

Refer to "Editing Alerts" in the *Studio Guide* for information on how to define conditions for evaluating alerts, such as the above where an arithmetic expression ([Number sold] * [unit price] is 1500 or less) is used, using the Analytics Studio.

## 4.3.4　Evaluating Alerts (Scheduled Evaluation)

In the following example, alert messages are output by setting up alert rules for evaluating the schedule of single events.
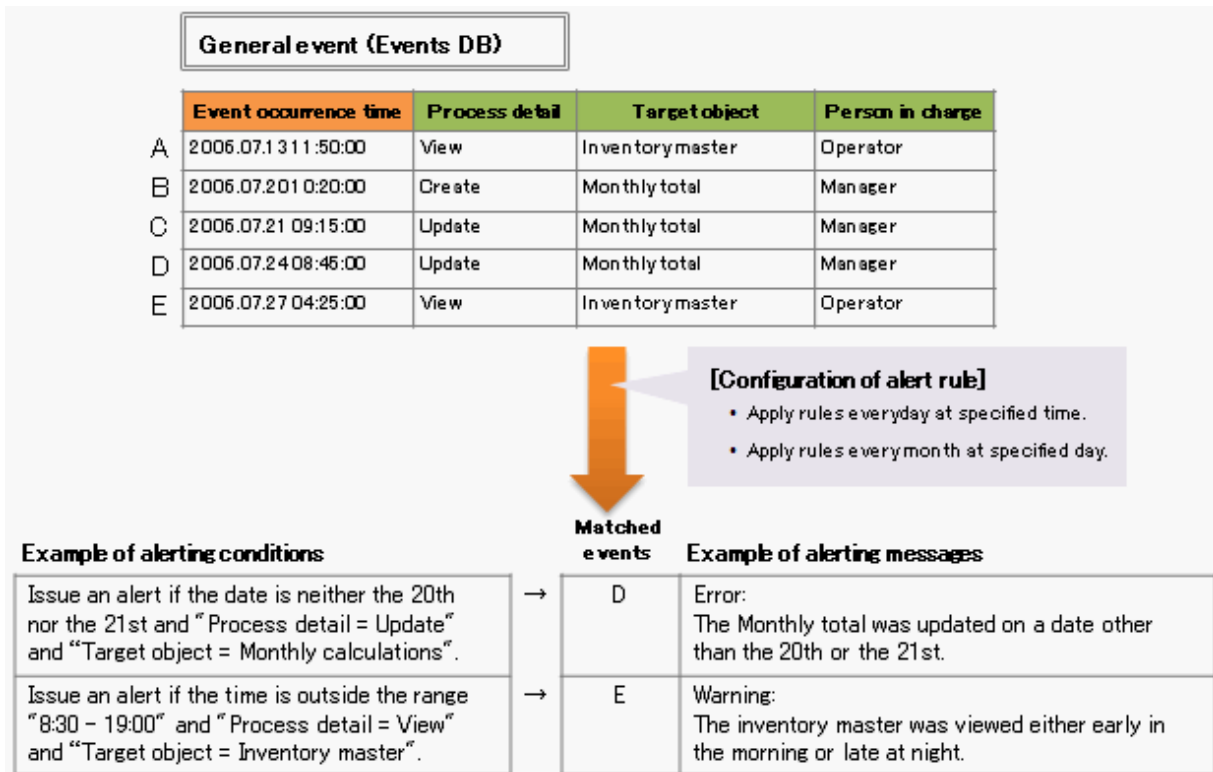
Events such as D and E can be detected by setting up alert rules as shown above. Alert messages will be displayed in the **Alert View** of the Dashboard.

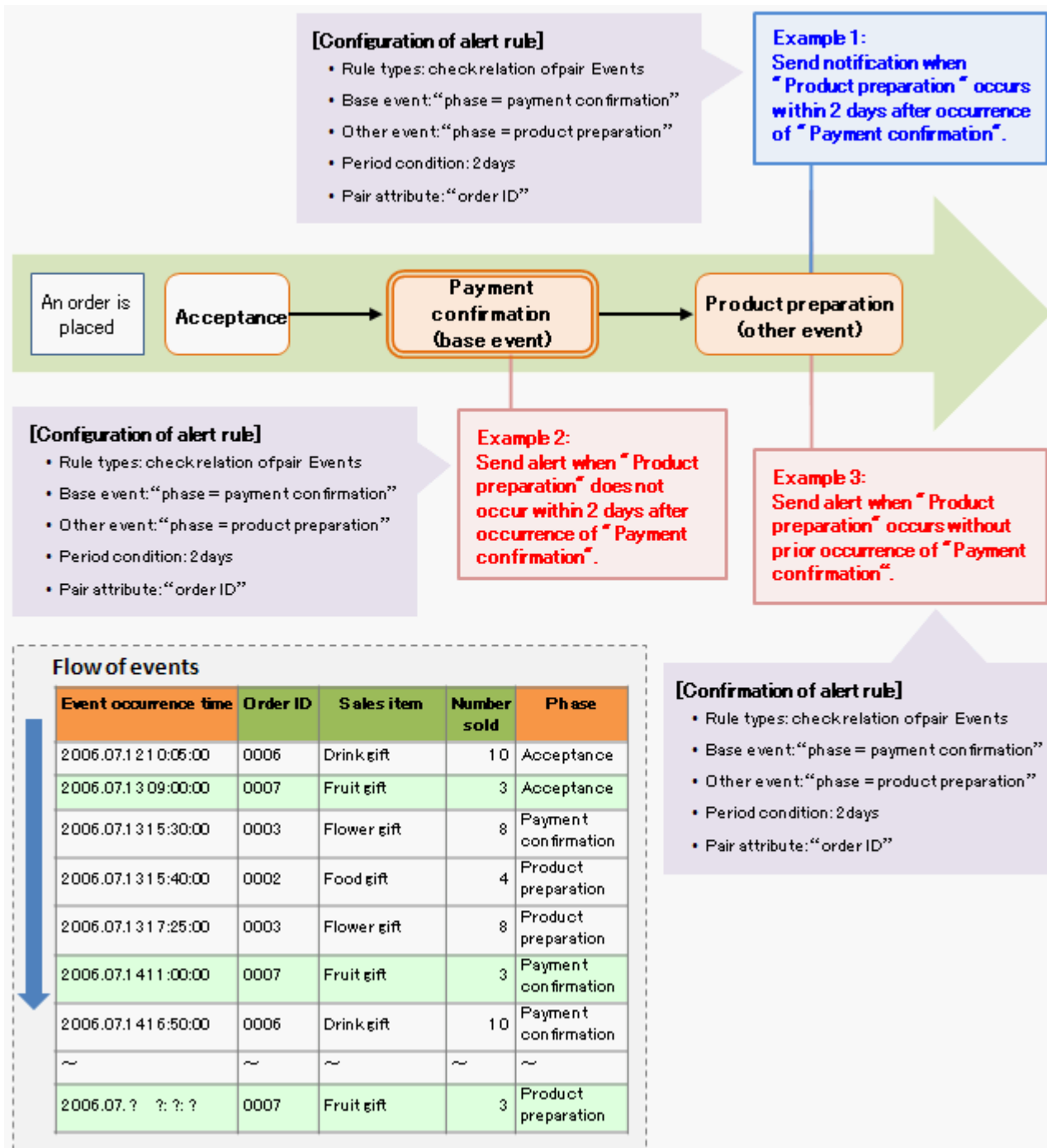## 4.3.5 Evaluating Alerts (Parameter Evaluation for Pair Events)

In the following example, alert messages are output by setting up alert rules for evaluating the parameters of pair events.
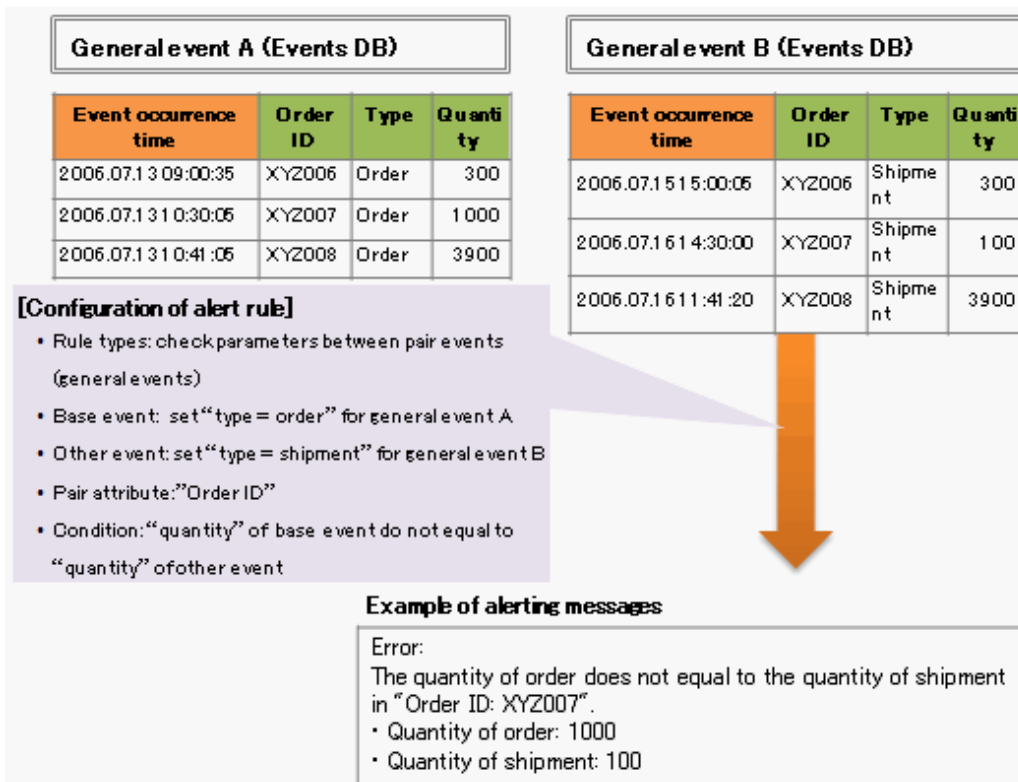
The attribute data values of two events can be compared as shown above by setting up "Order ID" as the pairing attribute and "'Quantity" of the base event is not equal to "Quantity" of other event'" as the determination rule.

4.3.6 Evaluating Alerts (Order Evaluation for Paired events)

In the following example, alert messages are output by setting up an order evaluation alert rule for paired events.

[Configuration of alert rule]
- Rule types: check relation of pair Events
- Base event: "phase = payment confirmation"
- Other event: "phase = product preparation"
- Period condition: 2 days
- Pair attribute: "order ID"

Example 1:
Send notification when "Product preparation" occurs within 2 days after occurrence of "Payment confirmation".

An order is placed → Acceptance → **Payment confirmation (base event)** → **Product preparation (other event)**

[Configuration of alert rule]
- Rule types: check relation of pair Events
- Base event: "phase = payment confirmation"
- Other event: "phase = product preparation"
- Period condition: 2 days
- Pair attribute: "order ID"

Example 2:
Send alert when "Product preparation" does not occur within 2 days after occurrence of "Payment confirmation".

Example 3:
Send alert when "Product preparation" occurs without prior occurrence of "Payment confirmation".

[Confirmation of alert rule]
- Rule types: check relation of pair Events
- Base event: "phase = payment confirmation"
- Other event: "phase = product preparation"
- Period condition: 2 days
- Pair attribute: "order ID"

Flow of events

Event occurrence time	Order ID	Sales item	Number sold	Phase
2006.07.12 10:05:00	0006	Drink gift	10	Acceptance
2006.07.13 09:00:00	0007	Fruit gift	3	Acceptance
2006.07.13 15:30:00	0003	Flower gift	8	Payment confirmation
2006.07.13 15:40:00	0002	Food gift	4	Product preparation
2006.07.13 17:25:00	0003	Flower gift	8	Product preparation
2006.07.14 11:00:00	0007	Fruit gift	3	Payment confirmation
2006.07.14 16:50:00	0006	Drink gift	10	Payment confirmation
~	~	~	~	~
2006.07.? ?:?:?	0007	Fruit gift	3	Product preparation

This is an example of a business activity where events are collected in three phases ("Acceptance", "Payment confirmation", and "Product preparation") after orders are placed, and then stored in the Events database. This section explains what happens when alert rules are set up for the "Payment confirmation" and "Product preparation" events of this business activity.

Example 1: Issue a completion report if "Product preparation" occurs within 2 days of "Payment confirmation"

This alert rule is used to confirm that the business activity has been processed normally.

An alert message advising of completion is output if the "Product preparation" event occurs within two days of the "Payment confirmation" event to which it is being compared. Comparisons are made for the same order by specifying "Slip ID" as the group attribute.

Example 2: Issue an alert if "Product preparation" does not occur within 2 days of "Payment confirmation"

This alert rule is used to detect business activities that have been delayed.

An alert message advising of the delay is output if the "Product preparation" event does not occur within two days of the "Payment confirmation" event to which it is being compared. Comparisons are made for the same order by specifying "Slip ID" as the group attribute.

Example 3: Issue an alert if "Product preparation" occurs without "Payment confirmation" having been confirmed

This alert rule is used to detect whether the business activity has been processed incorrectly.

An alert message advising of a business activity error is output if a "Product preparation" event occurs without a preceding "Payment confirmation" event. Comparisons are made for the same order by specifying "Slip ID" as the group attribute.

4.4   Overview of Chart Display

4.4.1   Chart Display (Basic)

In the following example, the events stored in the Events database are displayed as a graph.
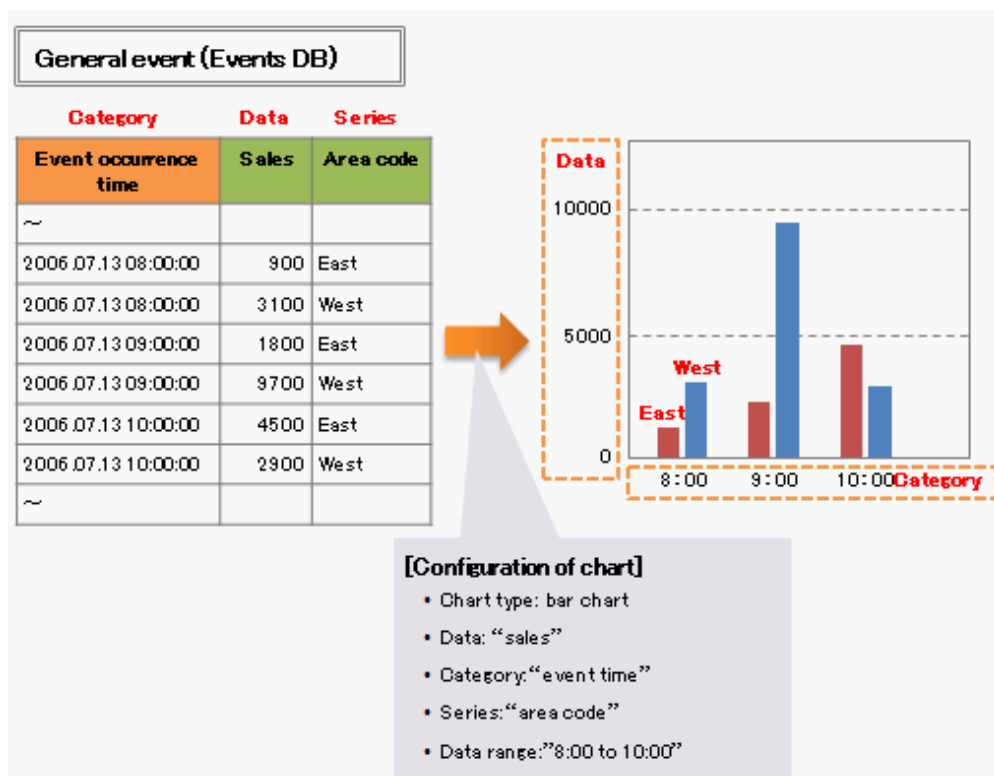
To display this type of graph, select "Bar graph" in the chart display settings, and specify "Total sales" as the data to be graphed, "Event collection time" as the item, "Area code" as the series, and "8:00 - 10:00" as the data collection period. The total sales for the KANTO and KANSAI areas will be displayed as bar graphs for each event, making it possible to see trends in total sales at a glance.

4.4.2   Data Processing

A variety of data is processed to produce the graphs used in chart displays. The following shows these processes.

Business Event

Event Collection datetime	Region	Product ID	Stock
2011.11.15 06:15:00	East	CODE0004	20
2011.11.15 07:15:00	East	CODE0002	80
2011.11.15 08:15:00	West	CODE0003	37
...	...	...	...
2011.11.15 10:15:00	West	CODE0002	42

[build data]
-Extract attribute used

[controller calculate Event data]
-Statistical event 0 plot control
-arithmetic operation
-calculation of attribute for same category value
-calculation by time series

[event data conversion]
-compare by time series(optional)
-control plot of series(optional)
-control plot of category(optional)
-sort data by category
-sort by series

[generate subgraph]
-calculate event data
-sort by series
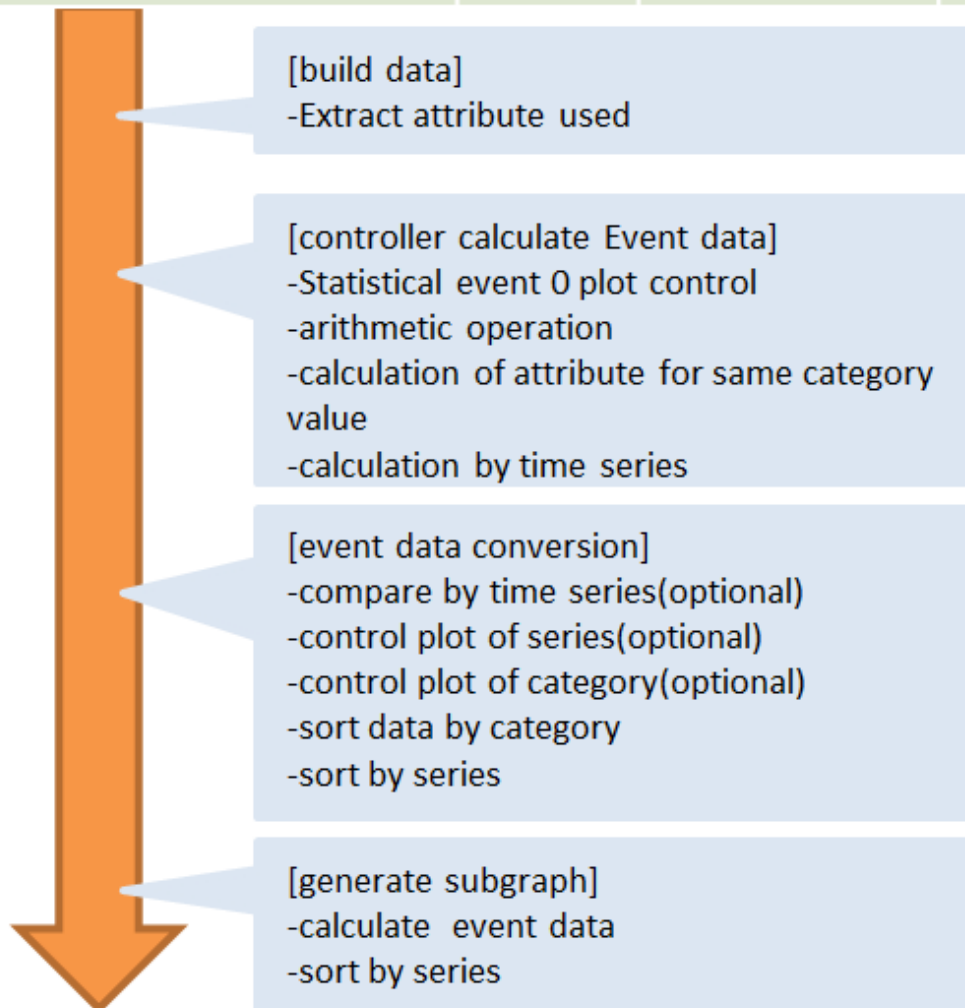
Display chart
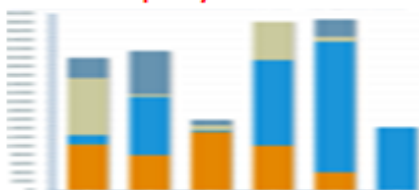
1. Retrieve event data from the Events DB

   Find the data in the Event DB using period and filter settings. The search results are then retrieved as event data. In the example the period is "XX", so all event data in this range is retrieved.

   🅿 Point
   ....
   You can avoid drops in performance by setting the period and filter so the number of search records displayed is limited to about 1000.

   Note that this is just a guide. Take your system into account when determining this value.
   ....

2. Build the dataset

   If the event data retrieved is field, series, data property, or an event table, a dataset is created by extracting the data of the specified property. If a sub-graph has been specified, it is extracted at this point.

   In the example, the item attribute "Event collection data", the series attribute "Region code", and the data attribute "Sales" are extracted.

3. Adjust and calculate event data

   This is where the data is adjusted and calculated. Where field values are duplicated, they are calculated according to the method specified.

   If a basic arithmetic operation has been specified in the data properties, that operation is performed. If there is event data with the same date in a field, it is calculated with the statistical operation specified.

4. Convert event data

   This is where other optional features can be used to adjust the plotting, etc.

5. Generate sub-graph data

   Sub-graphs are processed in the same way as the base graphs.

The above flow shows the processing that data in the Events DB must go through to be displayed as a chart. Charts may be displayed by the above processing, or they may be displayed as the result of statistics events that process the data beforehand. Each method has its merits and demerits. Use the method best suited to your purposes.

Processing data when about to display a chart has the following merits and demerits:

Merit

   The most up-to-date data is used.

Demerit

   Display may take a long time or the system may become unstable if the volume of data is great.

Using statistics events to display a chart has the following merits and demerits:

Merit

   The load caused by display of the chart is reduced by preprocessing the data.

Demerit

   The data must be processed beforehand so the displayed data will not be the most recent.

🅿 Point
....
The frequency of data updating must be taken into account when displaying charts.

When creating a chart of daily sales where the sales data is updated once a day, the data only needs to be processed once a day. By accumulating the data, as it is updated, you can reduce the load on the chart display.

With stock data, that may change minute by minute, the most recent data can be displayed by processing at the time of display. All that needs to be done to reduce the load caused by display is to make the period only as long as necessary.
....

4.4.3 Chart Display (Advanced)

In the following example, the events stored in the Events database are displayed as a graph, together with the graph from the previous month and threshold values.
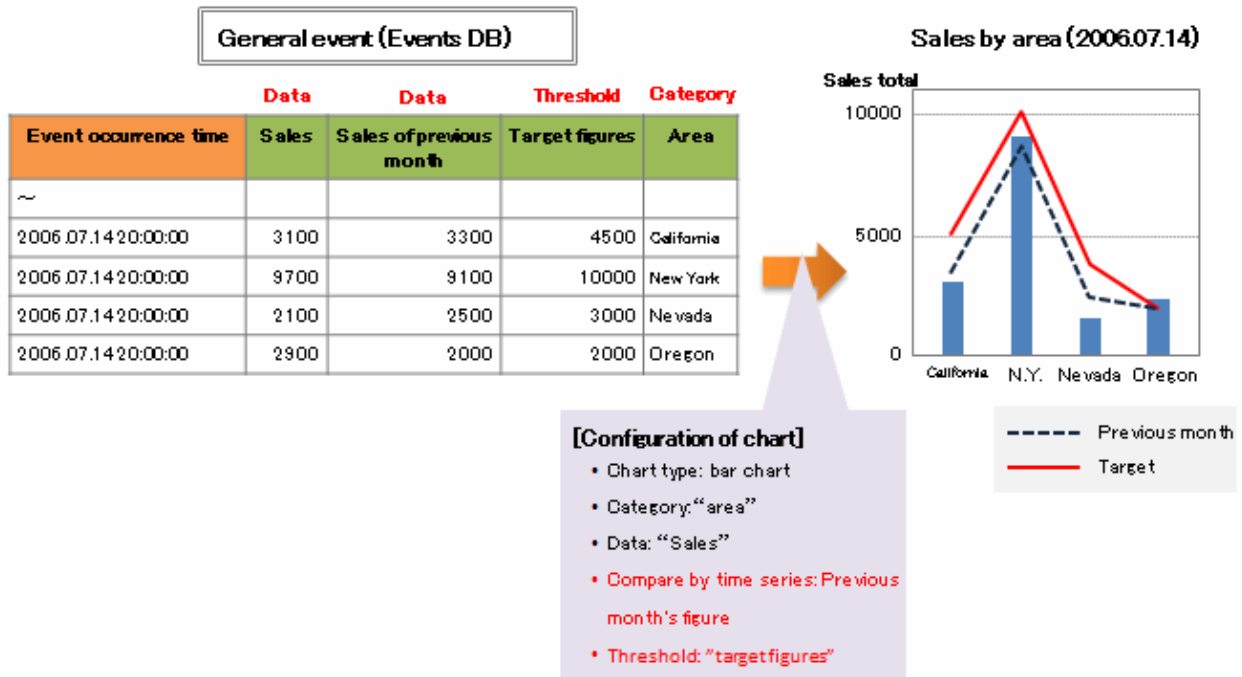

To display this type of graph, select "Bar graph" in the chart display settings, and specify "Total sales" as the data to be graphed, "Area" as the item, "Previous month's figure" as the superimposed data, and "Target line" as the threshold display. This graph makes it possible to compare the current month's figures with the target line and the previous month's figures at a glance.

4.4.4 Chart Drill-Down Display

The following example shows how to drill down through a chart display.

Sales chart by area

East2:1200
West1:2000
West2:1000
East1:1500
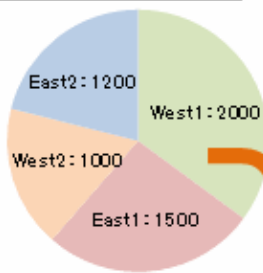
Click "West1"

Configuration of chart

Chart type	Pie chart
Category	Area
Series	–
Data	Sales

General event

Sales	Area
2000	West1
1500	East1
1000	West2
1200	East2
~	~

Display chart filtered by "Area = West1".

Sales chart by prefecture

California
Nevada California Oregon Nevada Oregon Nevada Oregon

5/1   5/2   5/3
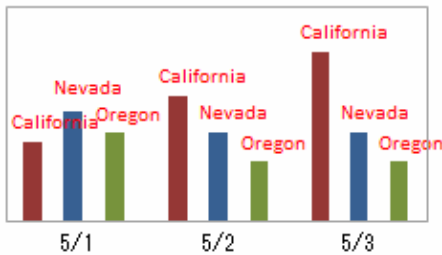
Configuration of chart

Chart type	Bar chart
Category	Event occurrence time
Series	Prefecture
Data	Sales

Configuration of drilldown

Parameter to be passed from parent chart	Area (Category)
Corresponding attribute name	Area
Parameter name to be passed to child chart	PARAM 1

General event

Event occurrence time	Sales	Prefecture	Area
2007.05.01	250	Tokyo	East1
2007.05.01	180	Florida	East1
2007.05.01	130	Georgia	East1
2007.05.01	200	Nevada	West1
2007.05.01	250	Nevada	West1
2007.05.01	200	Oregon	West1
2007.05.01	190	MaryLand	East2
~			
2007.05.02	340	N.Y.	East1
~			

In this example, the "Chart of sales by region" chart has been set as the parent and the "Chart of sales by prefecture" chart has been set as the child. Clicking on the "Kansai" region name in the parent chart causes the display to drill down to the chart for the Kansai region. The general events specified in the chart settings do not have to be the same, different general events can be specified for the parent and child charts.

4.5   Overview of Display Control in the Dashboard

Dashboard Groups are used by the dashboard to connect users and profiles in order to control what is displayed.

Access permissions are granted by first defining Dashboard Groups and then associating users and profiles with these groups.

There is a "guest" group in the system. Users that do not belong to any other group will be allocated to the guest group. A user cannot belong to the guest group and another group at the same time.

Dashboard Group A   Dashboard Group B

Dashboard Profile X

Dashboard Profile Y

Dashboard Profile Z

a   b   c   d   e   f   g   h   i

User: a,b,c
Profile: X,Y

User: d,e,f
Profile: X,Y,Z

User: g,h,i
Profile: Y,Z

Guest Group

Dashboard Profile G

Cannot be assigned to both Guest Group and other Group at the same time.

j   k
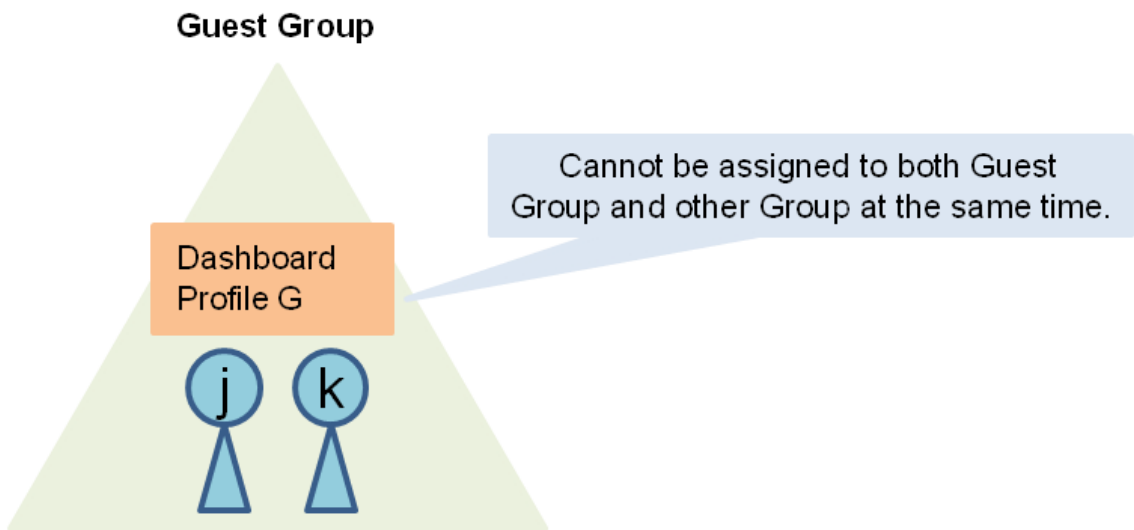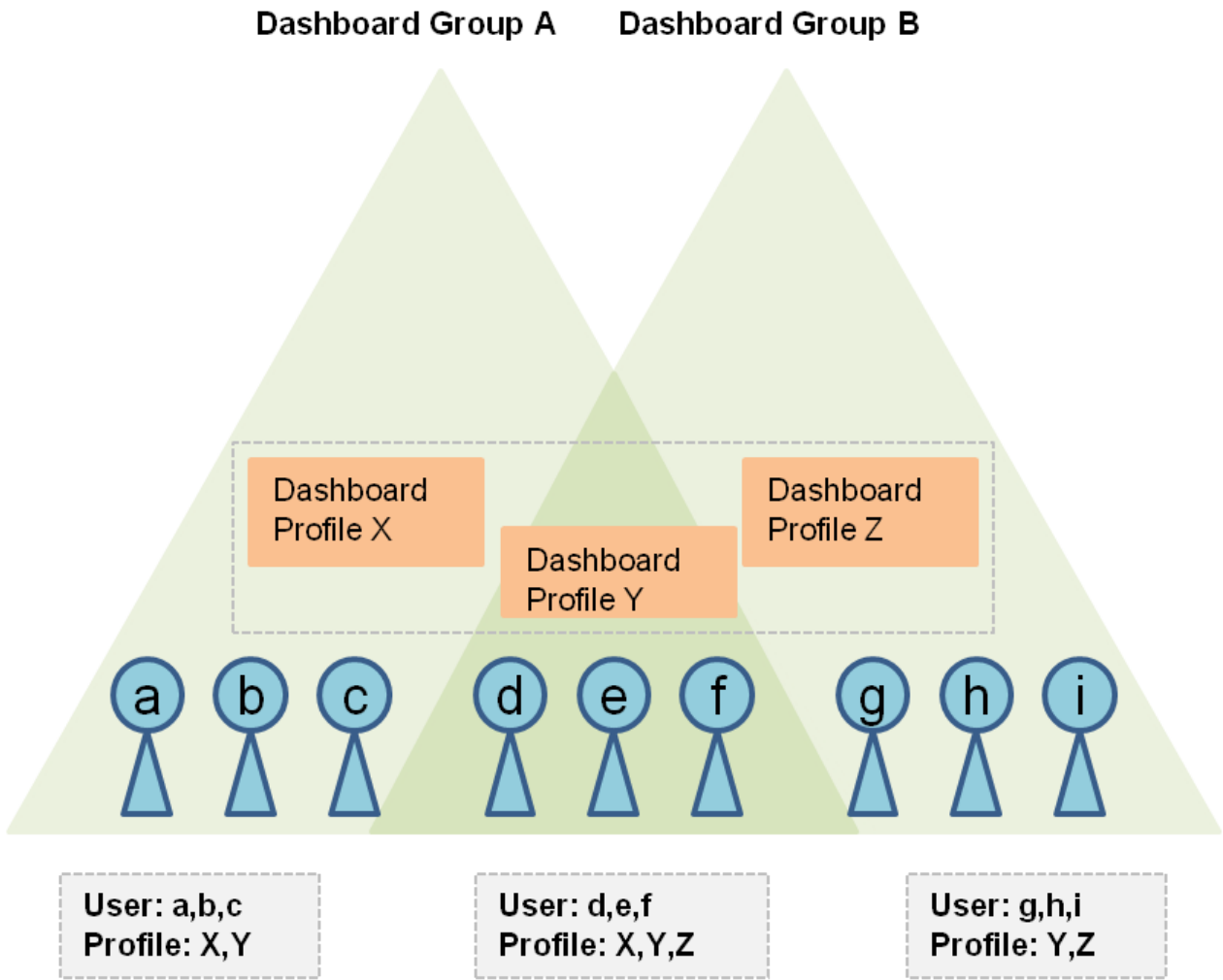
4.5.1   Example of Display Control

This example uses production processes and sales processes. Create "Production Processes" and "Sales Processes" Dashboard Groups. By associating Production Department User A with "Production Processes" and Sales Department User B with "Sales Processes", Production Department User A will be able to view only production data. Similarly, Sales Department User B will be able to view only sales data.

Dashboard Group

Product Management Business	Sales Business
-Product Management Profile -Production volume transition chart -Production volume alert	-Sales profile -Monthly sales performance chart -Sales transition chart

Production volume transition alert

Display Dashboard

Product Management User A

Product Management User A
-Product Management Business

Monthly sales performance chart   Sales transition chart

Sales User B

Sales User B
-Sales business group

4.6 Direct Display

4.6.1 External Filtering

By setting the appropriate request parameters when accessing the Business Reports view, the filtered data can be displayed in a chart.

This is an example of using HTML tags in a form to filter.

```
<form method="POST" action="http://hostname/ibpmm/dashboard/Welcome?
menuType=kpi&viewProfileId=VPR_AAA&chartLayoutId=DLT_BBB">
  <input type="hidden" name="conditionType" value="name" />
  <input type="hidden" name="key0" value="ProductName" />
  <select name="value0" multiple>
    <option value="DVD_media">DVD_media</option>
    <option value="plasmaTV">plasmaTV</option>
    <option value="cooler">cooler</option>
  </select><br/>
```

```
   <input type="submit" value="send" />
</form>
```
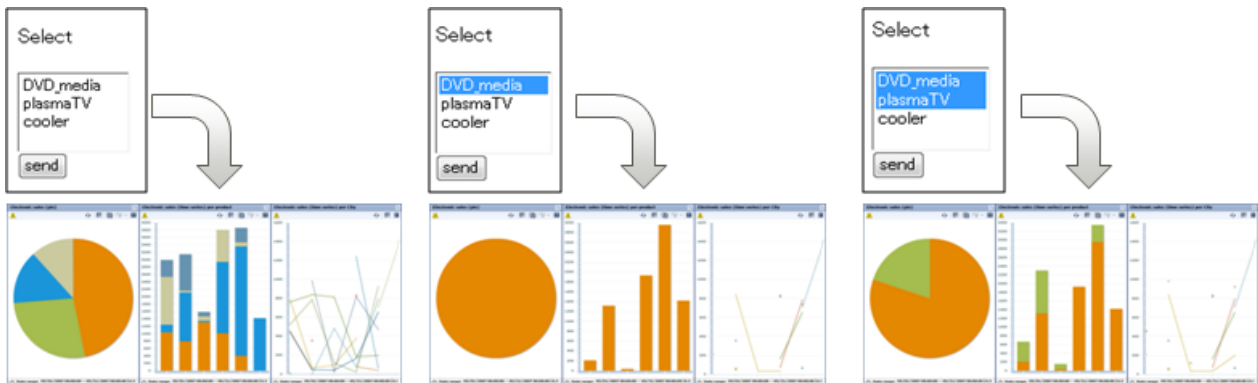

Nothing is selected in the example on the left so no filtering is performed on the displayed chart.

"DVD_media" has been selected in the middle example, so only the results for "DVD_media" is shown in the chart.

The example on the right has both "DVD_media" and "plasmaTV" selected, so data for both is displayed in the chart.

Refer to the Interstage Business Process Manager Analytics Dashboard / Output Guide for details.

4.7 Overview of Escalation

4.7.1 E-mail Example

Escalation is the process that is performed when an alert occurs. Here we will use e-mail as an example.

E-mails may be created in English and Japanese. A title and body text may be written for the e-mail. Information from the alert may be embedded in these.

New event data

Event Collection datetime	Region	Product ID	Stock
2011.11.15 10:15:00	East	CODE0005	20

Stock of new event is less tha 30

[Alert Rule]
-Condition: Stock < 30
-Alert type: Warning
-Level: 50
-Message: {Product ID} stock is less than 30.

Register event

[Action Rule]
-Recipient(s): manager@sample.jp
-Subject: Confirm stock of {Product ID}
-Body: Stock of {Product ID} is running short.(Stock:{Stock})
    Confirm immediately.

Alert triggered

[Mail Action]
-Recipient(s): manager@sample.jp
-Subject: Confirm stock of CODE0005
-Body: Stock of CODE0005 is running short(Stock:20)
    Confirm immediately.

Business Event

Event Collection datetime	Region	Product ID	Stock
2011.11.15 06:15:00	East	CODE0004	20
2011.11.15 07:15:00	East	CODE0002	80
2011.11.15 08:15:00	West	CODE0003	37
...	...	...	...
2011.11.15 10:15:00	West	CODE0002	42

4.8   Replacing Event Data

Set up label replacement definitions, a combination of values before and after replacement, to show the replacement values in the dashboard.

This is an example of replacing event data for display in the dashboard.

In this example, the product ID is replaced with the product name.

Label substitution data

key	value
CODE0001	television
CODE0002	recorder
CODE0003	refrigerator
CODE0004	washer

Substitute by registered label data

Event data

Events DB

washer: 100

refrigerator: 120

television: 250

recorder: 180

Business Event

Product Code	Number of sales
CODE0001	250
CODE0002	180
CODE0003	120
CODE0004	100
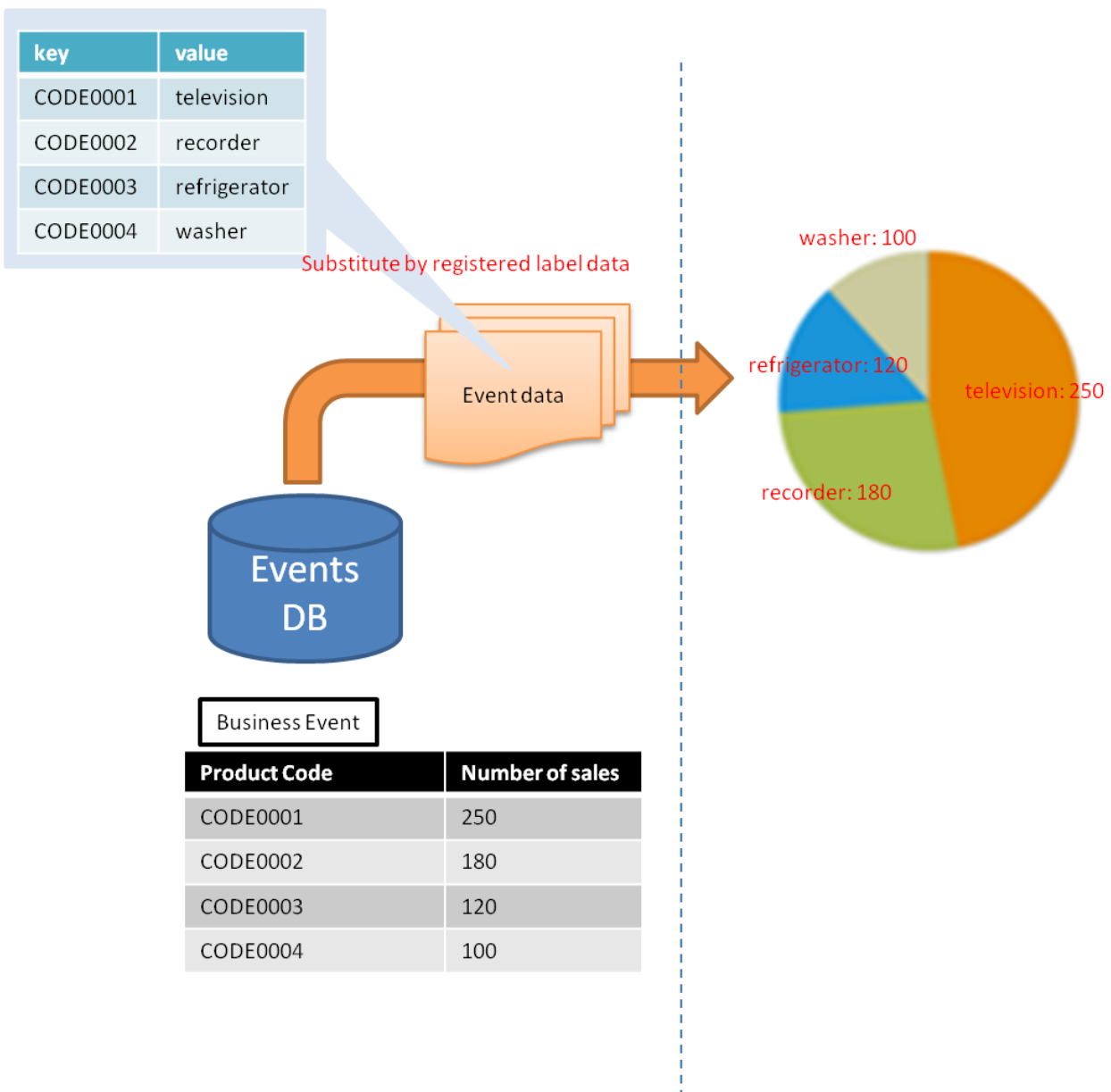
Define label replacement in the **System Settings > UI Management > Label Substitution** tab of the Management Console.

The definitions are set in Studio events for each attribute. The attributes thus set will be replaced for display when shown in the dashboard.

4.9 Personal Filter

This chapter describes a quick guide of Personal Filter feature.

4.9.1 Overview

Personal Filter feature allows showing only related data to individual users by setting filter rule per users. Also, each filter rule can be set whether user can temporarily remove them.

Following picture shows an example of how Personal Filter Condition works:

Personal Filter Condition

Condition Name	Event Attribute
Area Filter	Area Code
Product Filter	Product Code

Event Data

Area Code	Product Code
EAST	PROD1
WEST	PROD3
EAST	PROD4

User A's Applied Condition

Condition Name	Event Attribute Value
Area Filter	EAST
Product Filter	PROD1, PROD2

Display Dashboard

User A's Dashboard

Area Code	Product Code
EAST	PROD1
WEST	PROD3
EAST	PROD4

East Area User A

User B's Applied Condition

Condition Name	Event Attribute Value
Area Filter	WEST
Product Filter	PROD3, PROD4

Display Dashboard

User B's Dashboard

Area Code	Product Code
EAST	PROD1
WEST	PROD3
EAST	PROD4

West Area User B

In this example, User A and User B are applied Area Filter and Product Filter (Filter Rule: AND). In this situation, each user's Dashboard only shows the data which was matched each user's Applied Condition.

![Note icon] Note

Personal Filter Conditions are disabled by default. After created conditions, these conditions can be set to enable by editing Dashboard Profile, Alert Filter or Chart at Analytics Studio.

4.9.2   Usage

Following is a common step to configure Personal Filter Condition.

1. Create a new Personal Filter Condition

2. Apply the condition to user (Create Applied Condition)

3. Enable Personal Filter Condition at Dashboard Profile, Alert Filter or Chart editor

1. Create a new Personal Filter Condition

   To create condition, Management Console or Management Command is available. For small scale change, Management Console is appropriate to use. For large scale change, Management Command is appropriate to use.

   Followings step describes an example of when Management Console was used.

   1. After logging into Management Console, open **Create Condition** page

   ```
   User Management  >  Personal Filter Condition  >  Create Condition
   ```

   2. Input **Condition Name** and **Event Attribute**, then, click **Create** button

2. Apply the condition to user (Create Applied Condition)

After created Personal Filter Condition, you can apply the condition to users.

Following step describes how to apply condition to users:

1. Open **User List** page and click target user ID. Then, open Apply **Personal Filter Condition** tab

```
User Management  >  User List  >  (User ID)  >  Personal Filter Condition
```

2. Click target condition which you want to apply to user

3. Check **Allow Remove Filter** checkbox if you want to allow user to temporarily remove filter, input **Event Attribute Value**, then, click **Apply** button

3. Enable Personal Filter Condition at Dashboard Profile, Alert Filter or Chart editor

Following step describes how to enable created condition.

1. After logging into Analytics Studio, open the editor of Dashboard Profile, Alert Filter or Chart

2. In the editor, check **Use Personal Filter** checkbox at **Personal Filter Settings** area

3. For **Filtering Rule** radio button, if you want to use multiple conditions select **AND** if you want to only show data which matches all condition or select **OR** if you want to show data which matches any one of condition

Note
....
If you use **AND** for **Filtering Rule**, please apply all conditions which were set to the Dashboard Profile, Alert Filter or Chart to all users who have access to these elements. To avoid data to be shown to unintended users, no data will be shown to users until all necessary conditions applied to them.
....

4. Select conditions

5. Publish the element

Point
....
Personal Filter Condition setting of each definitions are applied as following priority

- Chart

- Alert Filter

- Dashboard Profile
....

For more details, please refer each manual.

For creating conditions and applying them, please refer "*Personal Filter Condition*" and "*Apply Personal Filter Condition*" of "*Management Console Guide*".

For Management Command, please refer "*Personal Filter Condition*" of "*Management Console Guide* ".

For enabling Personal Filter Condition, please refer "*Editing Dashboard Profiles*", "*Editing Alert Filters*" or "*Editing Charts*" of "*Studio Guide*".

4.9.3 Display Example

If Personal Filter Condition was used, user's Dashboards are like following.

When disabled Personal Filter Condition:

Area Code	Product Code
EAST	PROD1
WEST	PROD3
EAST	PROD4

When enabled Personal Filter Condition:

User A's Dashboard:

Area Code	Product Code
EAST	PROD1

User B's Dashboard:

Area Code	Product Code
WEST	PROD3

Also, if user was allowed to remove condition, user can do that from configure dialog in Dashboard.

Following is an example of removing Product Filter from User A's chart:

1. Show **Function Menu**
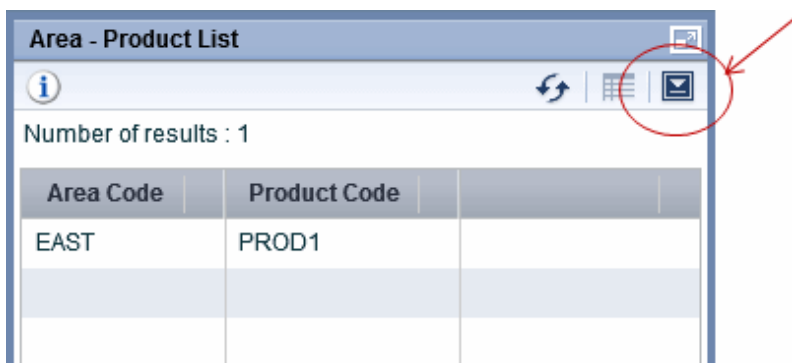
2. Select **Configure**
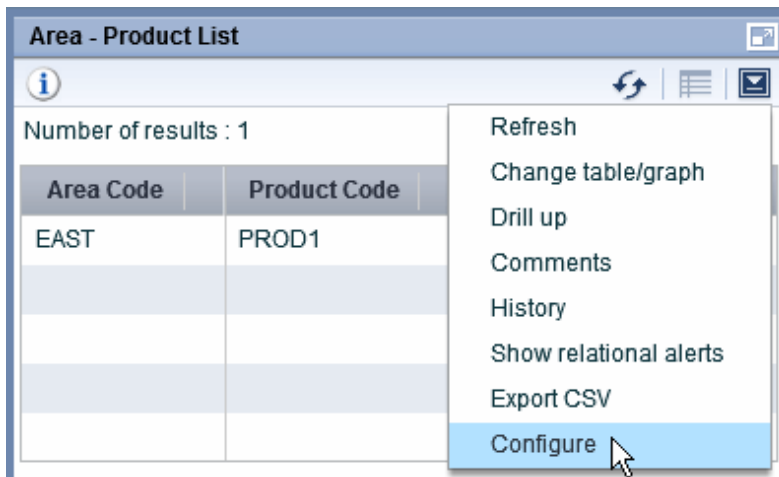
3. Open **Personal Filter** tab, then, remove check from Product Filter's checkbox.

   This tab's checkbox can control when Allow Remove Filter was set to Allow.
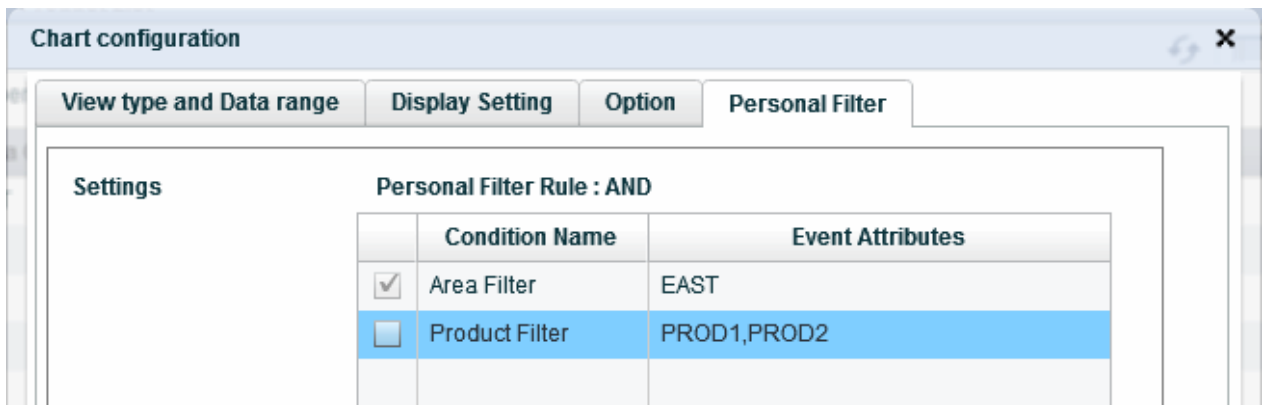
User A's Dashboard after removed Product Filter:

Area Code	Product Code
EAST	PROD1
EAST	PROD4

Chapter 5   Some Examples with Process Discovery

This chapter gives some examples with Process Discovery.

5.1   As-Is Analysis with Process Discovery

As-Is analysis with Process Discovery extracts logs from business systems, and then creates processes from the business data with the Process Generator. You can then use the following process analysis tools to find ways to improve the processes.

5.1.1   Comparison with Predicted Flows

- This feature allows you to see whether the processes are being performed as predicted.

- It compares the main parts of generated processes with the processes defined during design.
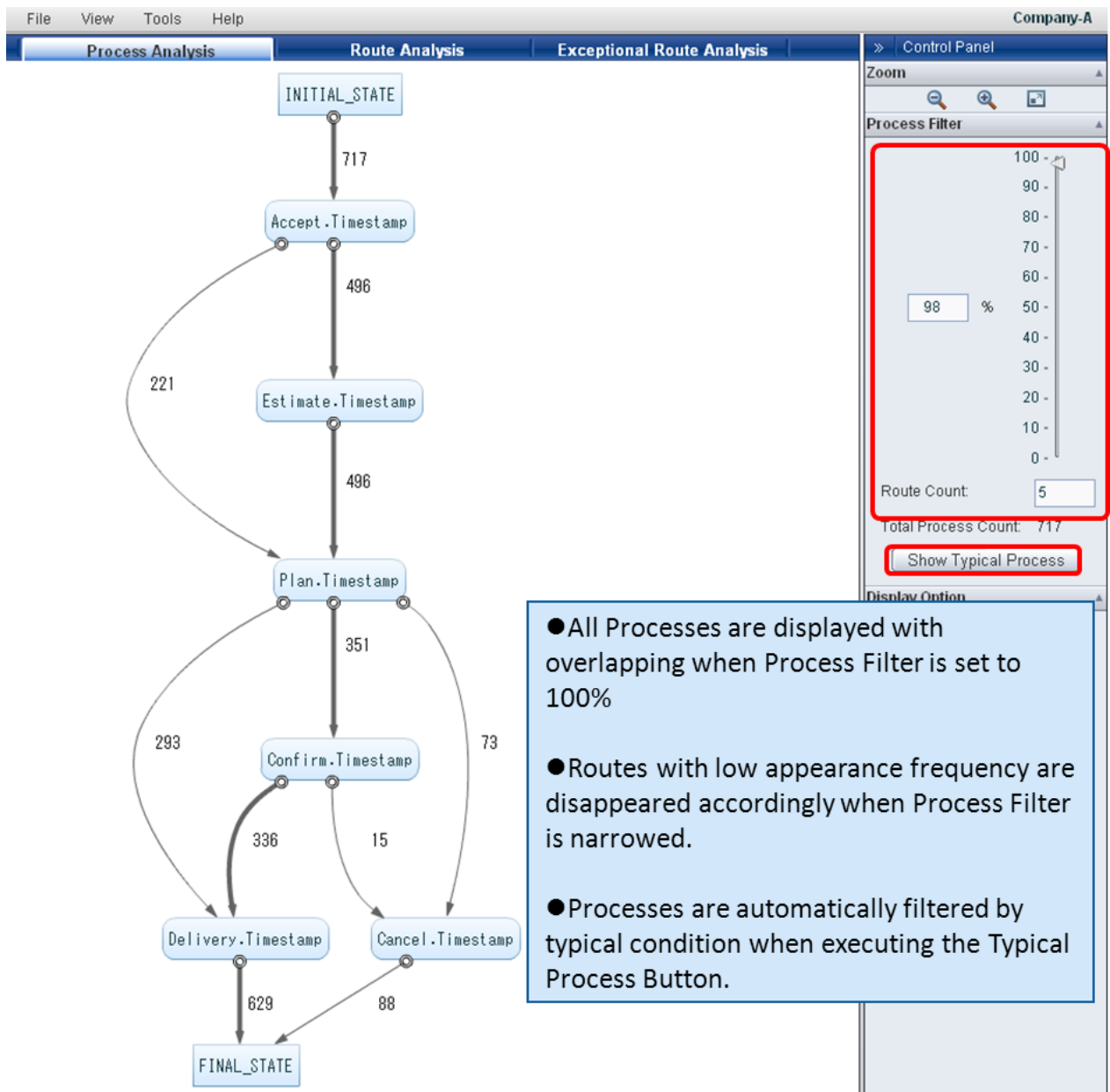
5.1.2   Detecting Exceptional Routes

Extract routes that include the specified transitions

- Details of process, which includes events with low appearance frequency, can be confirmed.

- Inadequate actions can be detected.

Extract routes that do not meet certain conditions

- Routes that do not meet some business governance conditions can be detected.

5.1.3 Detect Inefficient Processes

Detect repetitious processes

- Inefficient processes can be detected.

Detect process loopbacks

- Processes that should optimize some business procedures can be detected.

**Detect processes that take a long time**

- Processes that should improve some business procedures can be detected.
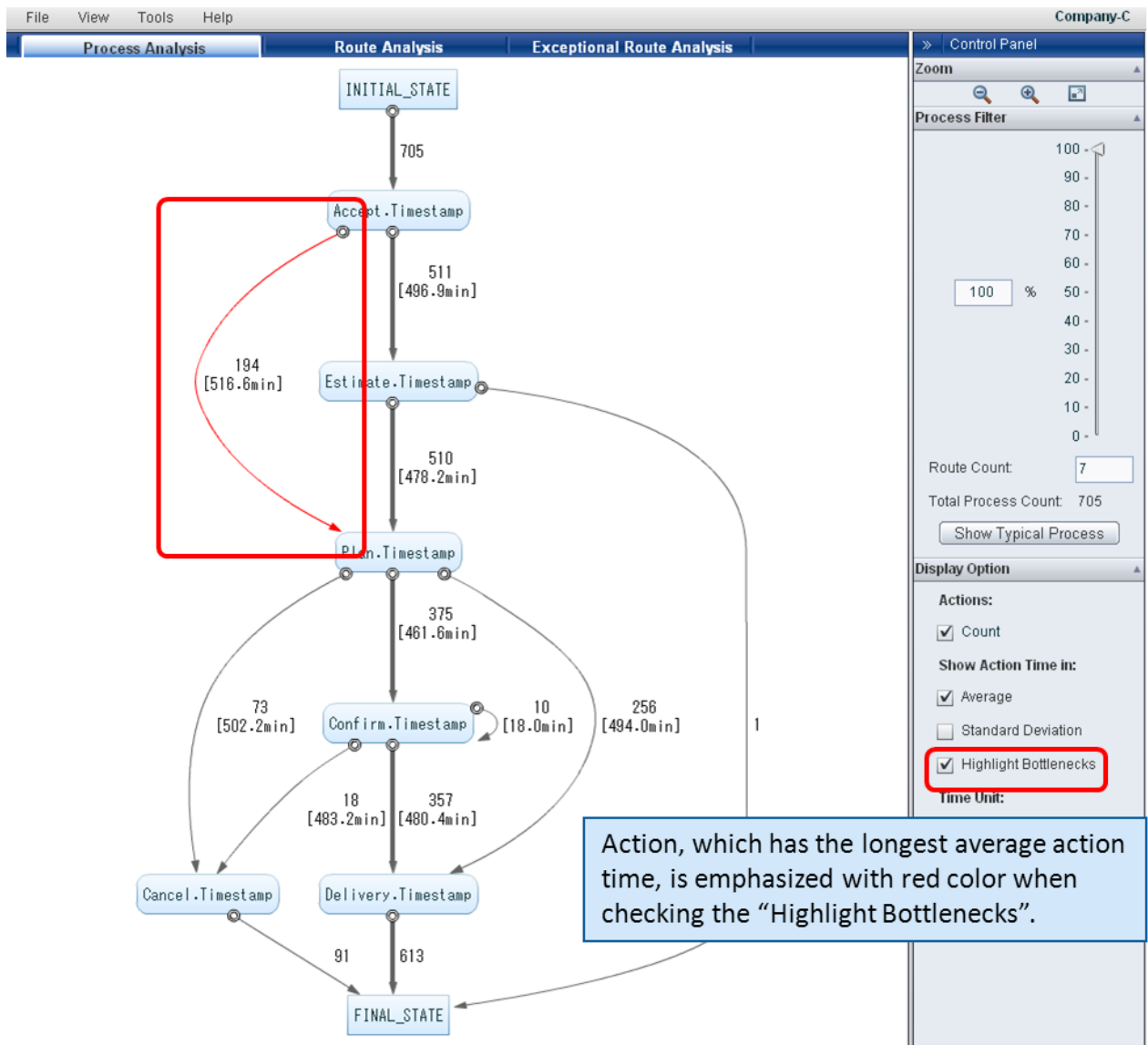
## 5.1.4 Displaying the Distribution of Action Time

Distribution of Action Time can be displayed as histogram.

If some properties are existed, property value is linked to each event. Using this, statistical information of action time can be displayed as well.
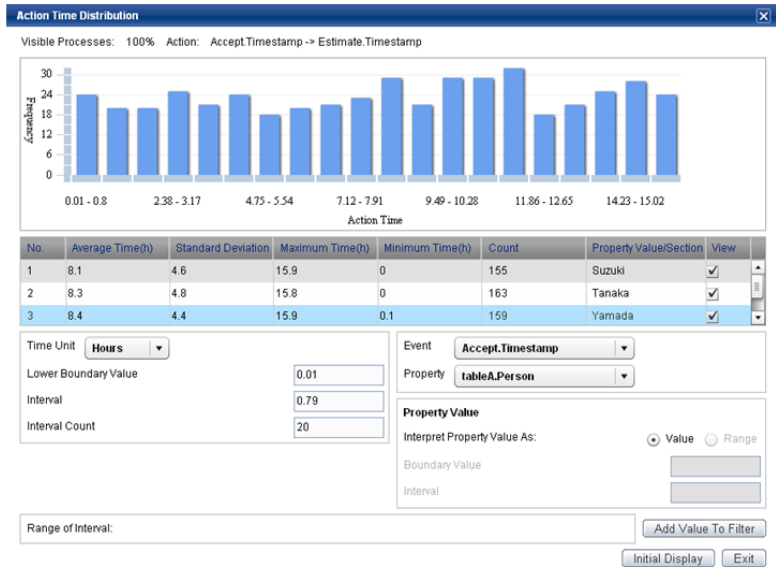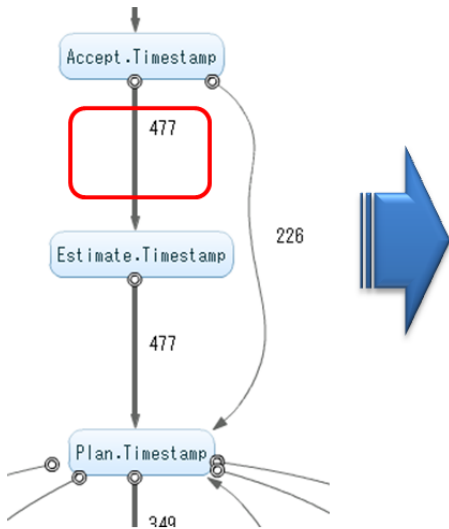
5.1.5 Display Property Distribution

If some properties are existed, property value is linked to each event. Using this, property distribution can be displayed as histogram.
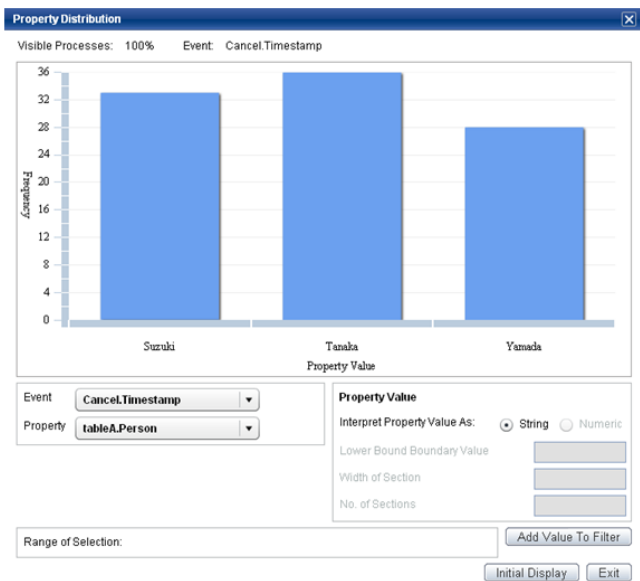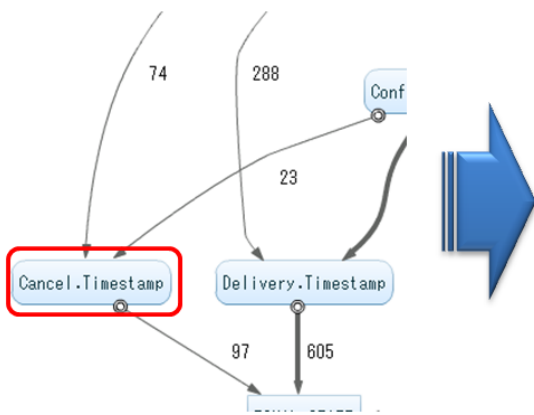
5.1.6 Looking at Trends in Routes

- According to left diagram, almost processes are concentrated on one main route. It means that improvement of business may be achieved if main route is optimized.

- According to right diagram, processes are diversified into various routes. It means that many patterns should be taken into consideration for improvement of busiess.
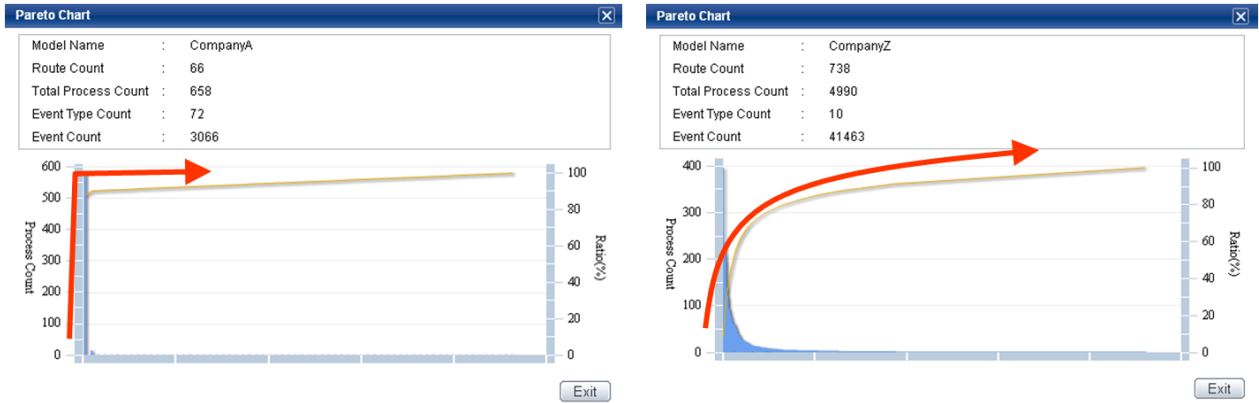
5.2 Fixed-point Analysis with Process Discovery

Detecting actions that do not comply, checking the results of improvement activities etc can be done by regular process generation using Analytics sensor.

5.2.1 Detecting Actions that do not Comply

[Process Analyzer]
An Alert is configured on this action as Exceptional Action.

[Dashboard]
Process Id is identified from Alert detail screen.

[Process Analyzer]
Detail of target Process is searched.

5.2.2 Checking the Results of Improvement Activities

[Process Analyzer]
A Chart is configured on this action as monitoring point.

[Dashboard]
A result which action time fell below the threshold can be confirm by KPI Chart on Dashboard.

Event Occurrence Time