

Ericsson Catalog Manager

Realize Higher Consistency for Faster Time-to-Revenue

Technical Product Description



© Ericsson AB 2014

All rights reserved. The information in this document is the property of Ericsson. Except as specifically authorized in writing by Ericsson, the receiver of this document shall keep the information contained herein confidential and shall protect the same in whole or in part from disclosure and dissemination to third parties. Disclosure and disseminations to the receiver's employees shall only be made on a strict need to know basis.



Contents

1	Product Description.....	5
1.1	Overview of Ericsson Catalog Manager's Key Product Modules	5
1.2	Overview of Ericsson Catalog Manager's Optional Modules	6
2	Ericsson Catalog Manager.....	7
2.1	Product Lifecycle Designer (PLD)	7
2.2	Product Data Management (PDM).....	10
2.3	Pricing	18
2.4	Discounting And Promotions.....	22
2.5	Taxation	23
2.6	Product Catalog and Service Catalog	23
3	Application Virtual Machine (AVM)	28
3.1	AVM Cache	28
3.2	AVM Cache Types	29
3.3	Certified Environments.....	30
3.4	High Availability Deployment Architecture	31
3.5	Deployment Sizing	34
3.6	Performance Management and Monitoring.....	39
3.7	Auditing and Event Logs	44
3.8	Archiving	46
4	Single Sign-On, Authentication and Password Management.....	47
4.1	Authentication and Password Management	47
4.2	User Privilege Management.....	50
5	Adapters	54
5.1	Competitive Advantage	54
5.2	Orga OPSC Gold Adapter.....	55
5.3	CSG Billing Adapter	58
6	Standards Adoption and Adherence.....	60
6.1	Solution Architecture	60
6.2	J2EE Architecture	61
6.3	Service Oriented Architecture	62
6.4	Web Services and Integration	63
6.5	BPM/Workflow	63
6.6	TeleManagement Forum: NGOSS, eTOM, TAM and SID	64
6.7	Frameworkx Compliance	64
7	Product Management	72
7.1	Software Release Strategy	72
7.2	End-Of-Life Policy	72
7.3	Documentation	73
8	Software Support Services	74
8.1	Introduction	74
8.2	Scope of Services	74
8.3	Software Warranty	75
8.4	Software Support Services	75



8.5	Terms.....	76
9	Trademarks.....	77
10	Disclaimer	77



1 Product Description

The Ericsson product portfolio is positioned as a set of enabling tools to help Communication Service Providers (CSP) manage their quote-cash supply chain whilst maximizing operational efficiency. Ericsson Catalog Manager's products comprise a set of pre-built software modules, which can operate independently or in combination with one other.

Ericsson Catalog Manager helps increase automation with product, service and resource management software that lets you quickly create new offerings from reusable components. For Communications Service Providers (CSPs), this results in rapid product introductions, increased revenue, fewer system updates, and lower operation costs.

Ericsson Catalog Manager comprises the following key modules:

- Catalog Designer
- Service Catalog
- Resource Catalog
- Application Virtual Machine

The following options are available:

- Product Catalog
- Product Lifecycle Designer
- Velocity Studio

1.1 Overview of Ericsson Catalog Manager's Key Product Modules

Catalog Designer

The Catalog Designer allows you to configure various aspects of your catalog, including configuring catalog privileges, and importing and exporting catalog data. It also allows you to create new and open existing projects, add and remove products, services, and resources, set tax rates, make item associations, and much more.

Service Catalog

Service Catalog supports TMF/TAM's technical product catalog, which is defined to provide support to fulfillment in terms of Service and Resource Provisioning and Service Order Management.



Resource Catalog

Resource Catalog supports TMF's definition of item types. The Resource Catalog allows for modeling products, services, and resources as items in the catalog.

Application Virtual Machine (AVM)

Application Virtual Machine (AVM) is Ericsson Catalog Manager's runtime engine for all metadata (Ericsson Catalog Manager's Application Markup Language (AML)). The AVM provides common services to metadata-based modules including a UI server, workflow engine, integration framework, rules engine, user authentication and authorization, and data persistence services. The AVM is the runtime framework of all Ericsson Catalog Manager applications.

1.2 Overview of Ericsson Catalog Manager's Optional Modules

Product Catalog

Product Catalog fully supports the TMF/TAM's commercial product catalog definition and provides overall product management support for TMF/TAM's product entity hierarchy. The product management functions include product definition; products offered; product business hierarchy; commercial product hierarchy; rules relating to offers including prerequisites and embellishment options, and relationships with other offers and parameters; availability rules; and validity period.

Product Lifecycle Designer (PLD)

Product Lifecycle Management (PLD) module provides the functionality to manage the lifecycle of items (products, services, offers, resources, etc.) from conception, through design and development, to service and retirement. PLD integrates people, data, process and systems to provide a single view of items. PLD currently supports the states defined by TMforum: Concept, Design and Develop, Deploy, Operate, Retire, and Manage Change. PLD provides Web-based user interface and a configurable lifecycle workflow out of the box.

Velocity Studio Designer (VS)

Velocity Studio Designer (VS) provides all of the functionality of Service Designer, as well as advanced UI form modeling that enables next-generation application design. VS is a critical enabler for end-user facing applications requiring an enhanced user experience, ease of navigation, etc.



2 Ericsson Catalog Manager

Catalog Manager provides a central runtime facility for cataloging the CSP's product and service portfolio. Catalog Manager houses the customer's product hierarchy (products, sub-products, features, etc.), and defines the parameters governing how the products are to be handled (allowable combinations, availability, workflow, etc.). In addition, Catalog Manager addresses key functions that current billing or CRM product catalogs cannot comfortably (or cost-effectively) handle, such as mapping commercial orders to technical orders, and interacting with multiple technology networks such as wireline, wireless, or IP-based offerings. Catalog Manager is further divided into two capabilities: the Product Catalog and the Service Catalog. Catalog Manager offers a Product Lifecycle Management (PLD) option as well.

2.1 Product Lifecycle Designer (PLD)

Product Lifecycle Designer (PLD) provides the ability to manage the end-to-end lifecycle of products from Idea to Retirement:



PLD provides:

- An end-to-end view of the product lifecycle
- A framework for managing and monitoring catalog changes
- A framework for collaborative modeling of products across workgroups
- Mechanisms for publishing and consuming catalog data to/from other systems, including third-party catalogs
- Mechanism for publishing catalog changes between IT environments, i.e. from development, through to acceptance testing and production

Ericsson Catalog Manager leverages its core platform to provide robust support for full PLD, including workflow, jeopardy and exception management, integration services, data transformation services and a work listing module with a full security and task-specific form modeling and business rules.

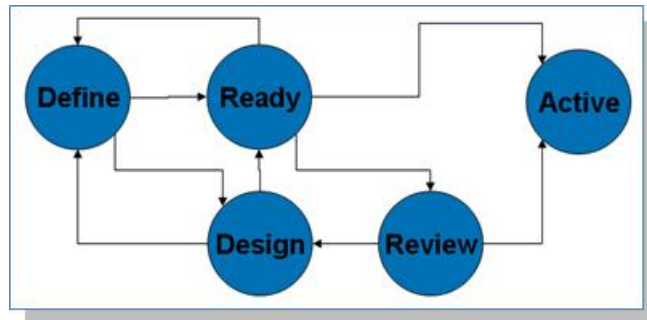


2.1.1

States

All Catalog Manager Entities have a state. The state is configurable and may be changed manually, through a state machine, and via workflow. Change Requests/projects are employed to group catalog changes for review and approval.

The default lifecycle within the catalog consists of 5 states as depicted below. These states, and the workflow used to drive them, are altered to meet the requirements of the product lifecycle as part of the solution deliverable via Ericsson Catalog Manager's Service Designer.



2.1.2

Projects and PLD Workflow

PLD employs a Project concept to group changes in the catalog. The first step in modeling a product, promotion or discount within the catalog is to establish a Project, which serves to collect all changes made in the catalog.

As the Project goes through the PLD lifecycle, the state of the underlying catalog changes are moved as a group through the catalog's life-cycle.

Project Detail

project Code * Status *

Name *

Label Effective Date *

Parent Project Owner

Description

Configuration

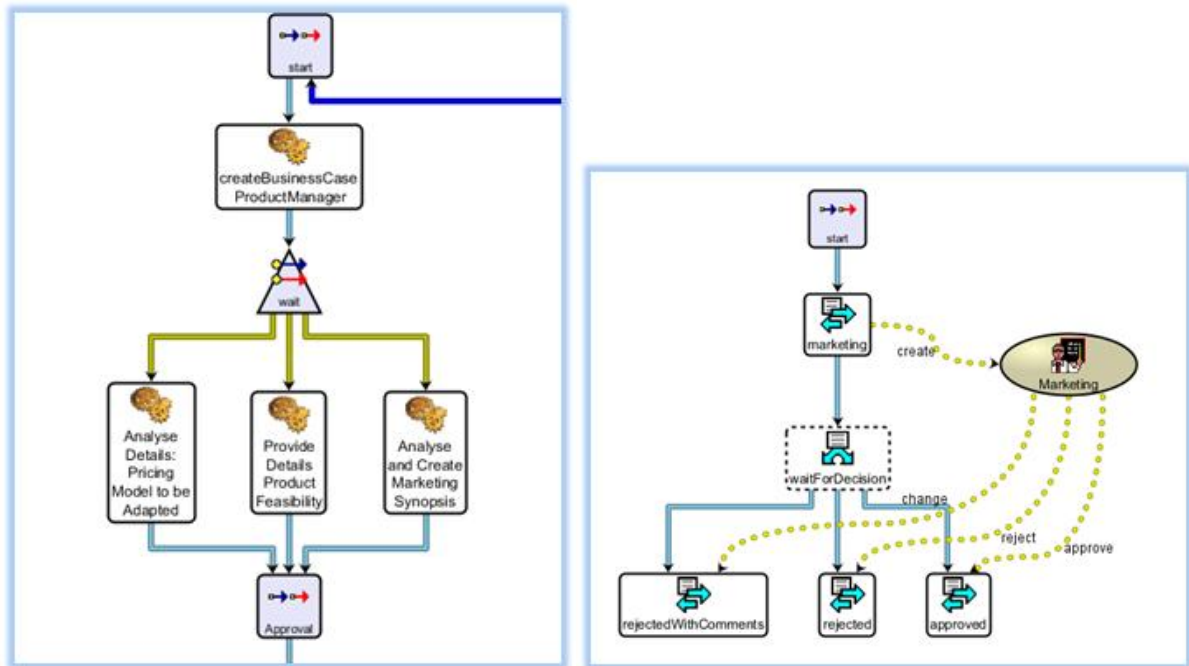
Templates

	Main	Evaluation	Conception	Definition	Build	Testing	Launch Product
Business Analyst	<input checked="" type="checkbox"/>			Due Date 04/28/2011			
IT	<input checked="" type="checkbox"/>			Due Date 04/28/2011			
Marketing	<input checked="" type="checkbox"/>			Due Date 05/18/2011			
Network Operations	<input checked="" type="checkbox"/>			Due Date 05/18/2011			
Pricing & Economics	<input type="checkbox"/>			Due Date			
Product Manager	<input type="checkbox"/>			Due Date			
Product Modeler	<input type="checkbox"/>			Due Date			
Revenue Assurance	<input checked="" type="checkbox"/>			Due Date 04/21/2011			



The Project provides a means to tailor the workflow associated with the project. Template functionality is provided enabling users to save a configuration for future use, either as is or as a template.

The template provides the ability to enable/disable individual phases within the macro PLD process, from Evaluation, through Conception, Definition, Build, Testing and Deployment. Within each phase, the actors / actions required are also selected. The phases and actors are configurable via workflow and can consist of macro-level workflow that covers a series of steps, or individual micro-flows that perform a single action. Examples of each follow:



The Ericsson Catalog Manager Service Designer provides an Integrated Design Environment (IDE) for the modeling of Telecom specific workflow in support of the PLD process. The workflow engine supports a full range of activities required for the PLD process including sequential and parallel tasks, manual and automated tasks, and exception, rollback and compensation activities. These capabilities are explored in more detail in their respective sections of this reference document.

The workflow is used to task users to perform catalog modeling roles within the catalog such as pricing tables, content management, discount management and approvals.



A major piece of the PLD workflow is the ability to publish catalog changes to other OSS/BSS systems that require catalog data, such as the fulfillment and billing systems. The integration of catalog data to/from third-party catalogs requires capabilities above and beyond simple Product creation, all of which are provided in the Ericsson Catalog Manager IDE:

- An IDE to model the interfaces
- Workflow to manage the packaging of the content into cross-supplier bundles
- Full domain based security model
- Workflow driven propagation to downstream systems
- A scheduling mechanism for batch based interfaces
- Work queues to assign tasks to various roles
- Alerts re: presence of new content, exceptions in publishing process
- Exception Management to handle incomplete / invalid content and content that could not be published to third-party systems
- Reconciliation processes to deal with the inevitable deviations between the catalogs

For details on Ericsson Catalog Manager's integration capabilities, please see the Integration section of this reference document.

2.2 Product Data Management (PDM)

The key design criteria in achieving rapid bundling of products, together with seamless fulfillment of these same bundles, is creating bundles from re-useable components. Ericsson Catalog Manager provides this in its catalog through the composition and inheritance models:

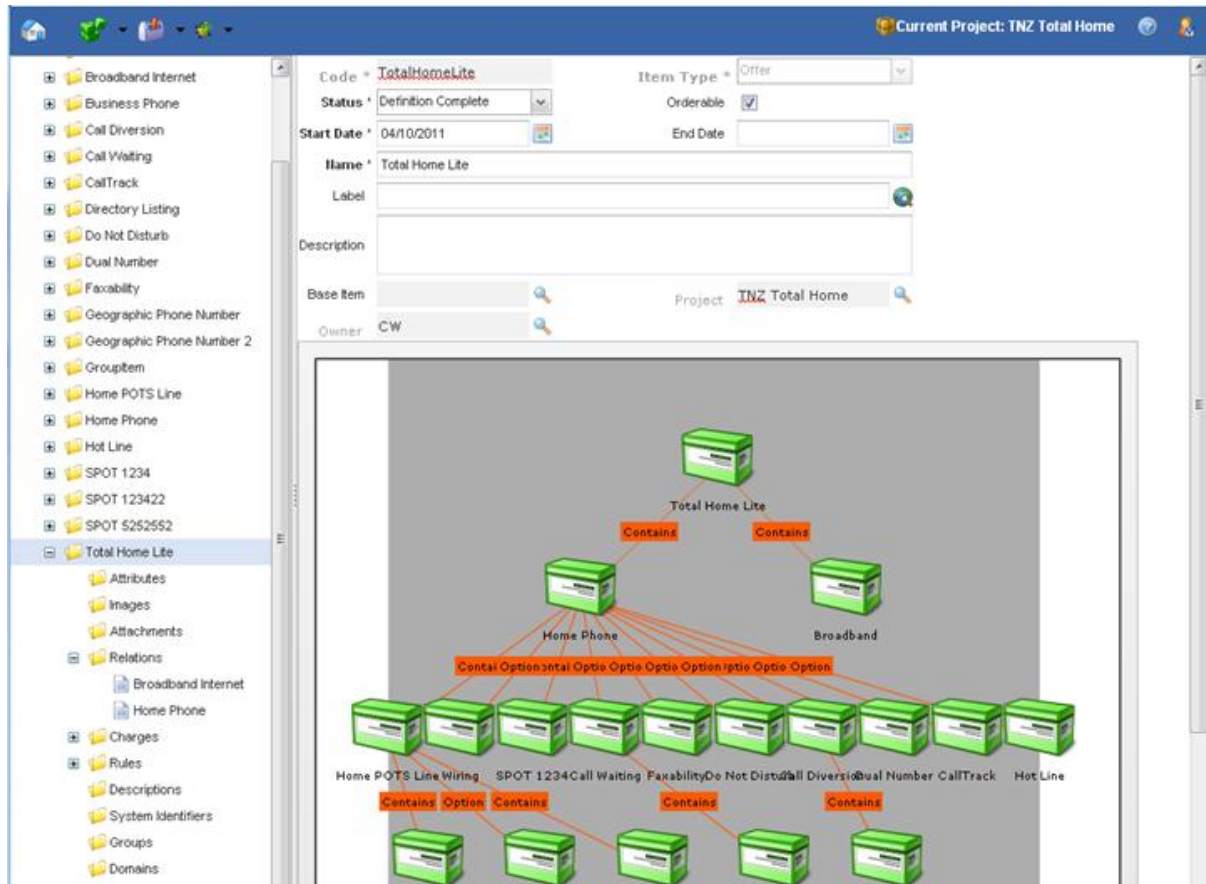
- Through composition, an item is defined as comprising one or more other items, each of which themselves may be composed of one or more items (to N levels).
- Through inheritance, an item is defined as an instance of another item, and inherits all of the "base" item's characteristics and composition.

In both cases, the item itself may modify the characteristics and settings of the items upon which it is based, and those that comprise it.

These models permit the definition of core services that are understood by other systems, such as a Pricing Item (IBIS) or SPOT code (ICMS), and recombine them to form new products and bundles without change to the underlying systems. Clearly where new capabilities are being introduced, changes to systems will be required. The power comes with products, bundles and promotions that consist solely in pricing and packaging variations on existing capabilities. These are able to be introduced rapidly with no change to the surrounding systems.



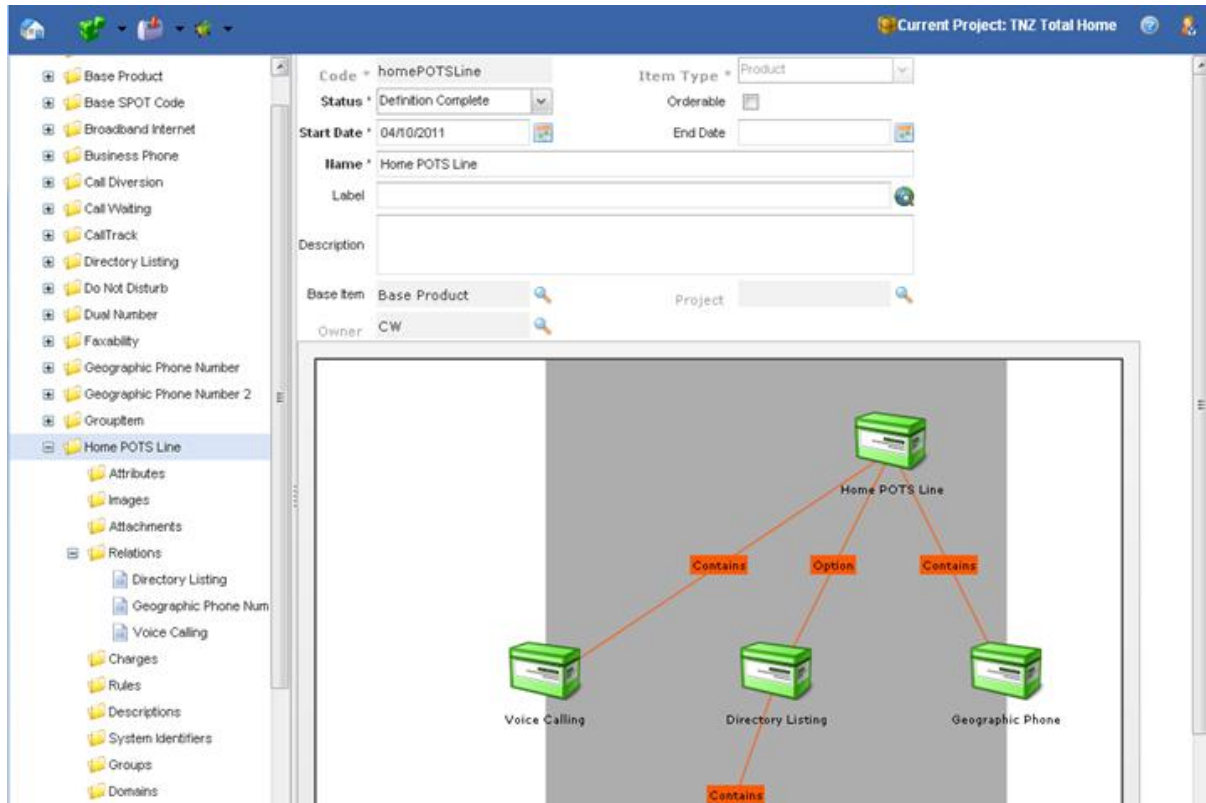
To facilitate describing capabilities, a sample product definition from Telekom New Zealand will be used as an example. Total Home is a bundle that consists of two Products, Broadband Internet and Home Phone. In Ericsson Catalog Manager, this is modeled by defining an Offer, “Total Home Lite” as the composition of the two underlying Products, as follows:



Each of the Broadband and Home Phone Products were created using a Base Product definition. This permits the definition within the Base Product of all elements and attributes that are required across all products, such as Revenue Reporting codes, so that they need not be redefined with each Product created.



Each Product is then built via composition by including all mandatory and optional Customer Facing Services, such as a Home POTS Line, and voice features such as Dual Number and Hot Line. In the case of Home Phone, Home POTS Line itself is composed of Voice Calling, Directory Listing and Geographic Phone.



Where optional items are concerned, a Group is defined that contains the cardinality rule, e.g. minimum 2 and/or no more than 5, that governs any business rules around the selection of options.

Items (at any level) may have Attributes defined that are either Properties, whose value is set by the catalog, or Attributes, whose value is expected to be defined/set during order entry (e.g. telephone number) or as part of fulfillment (e.g. assigned DLCI).

Relationships between items need not be constrained to a composition relationship. Relationships are user-definable and are typically created for specific purposes, such as “Implemented By” for the specification of the associated fulfillment workflow, and “Billed Through” for the specification of the associated billing item (itself modeled in the catalog).

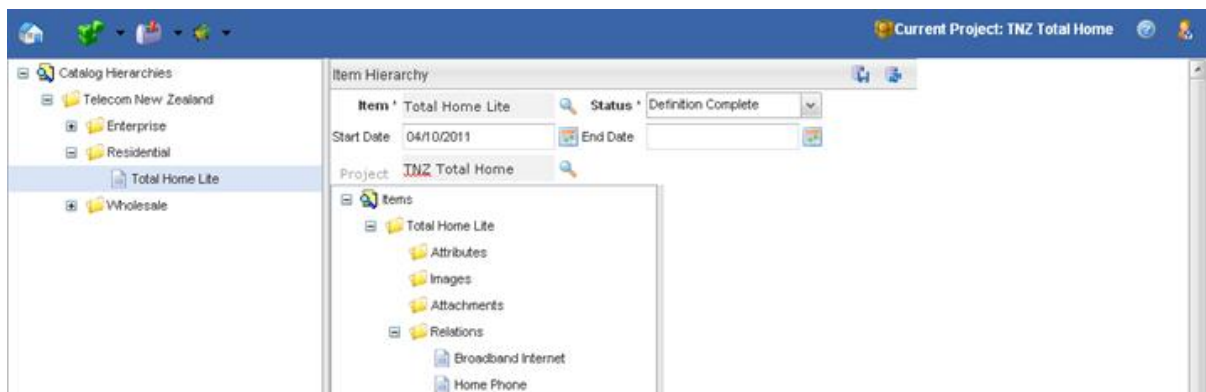


Pricing may be tied to any level of the hierarchy, e.g. at the bundle level, Product level, Customer Facing Service, or Group level. It should be noted that by default, pricing defined on an item will automatically be applied to any item that includes this item. Care should be taken in planning the catalog model and pricing to ensure that components that are re-useable not include pricing unless the pricing is known to be universal. Best practice dictates that re-useable components not carry pricing. If desired, an item may be created that includes the re-useable component and its pricing, thereby giving future modelers the option of using the component with or without pricing.

Similar logic applies to business rules, whether these rules are availability, eligibility, ranking, validation or attribute restrictions. Only universal business rules should be defined at the re-useable component level.

2.2.1 Catalog Hierarchy

Once modeled, orderable Items (Offers) are placed in one or more branches of the Catalog Hierarchy. Catalog Hierarchies are used to facilitate browsing through a large catalog of Products by categorizing the products. They are also used to simplify rule writing. For example, rather than adding an availability rule of "Retail Clients Only" to each and every retail product, it can be added once to the Catalog Hierarchy node itself, and will apply to all products in that branch of the tree.





2.2.2

Rules

The catalog includes a rules library, typically managed and defined by an IT person. This rules library represents re-useable business rules that are written once and leveraged by product Managers in the definition of their products. They may be passed parameters, and may access reference tables, further increasing their re-usability.

The screenshot shows a web application window titled "cwt_catalogRuleFinder". The main content area is titled "Catalog Rule Detail". It contains several input fields and a status dropdown. The "Rule Code" field is filled with "isresidential" and the "Status" dropdown is set to "Definition Complete". The "Name" field is filled with "isResidential". The "Start Date" field is filled with "04/12/2011" and the "End Date" field is empty. The "Rule Type" dropdown is set to "Condition". The "Label" field is empty. The "Description" field is empty. The "Project" field is filled with "TNZ Total Home" and the "Owner" field is filled with "CW". The "Rule" field is filled with the code "return theContext.typeofclient == 'RES';".

Business rules operate under a user-definable "Context". This context consists of a set of characteristics that define the current query, including the client's location, market segment, serviceability test result, and payment status. Also available to the rules are the current contents of the order, and the installed services for the client.

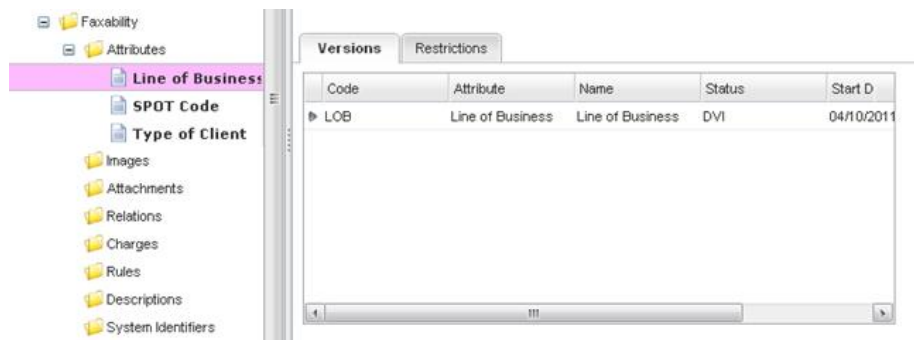
Modelers using the rules pick them from a list. For example, tying the Residential node in the Catalog Hierarchy to the rule "isResidential" restricts all products within the Residential branch of the Catalog Hierarchy to retail customers.



2.2.3 Versioning

The catalog provides full versioning capabilities. An item's definition at any point in time is defined as the collection of relationships defined at that point in time. In other words, items themselves are not versioned. Each attribute, relation, image, attachment and business rule of an item is versioned independently. Versioning also applies to Information Models, Catalog Rules, Charge Types, Attributes and all other stand-alone entities managed by the catalog.

The version employed is determined by the effective dates for each version defined.



2.2.4 Modeling Policies, Contracts, Order Decomposition and Orchestration Rules

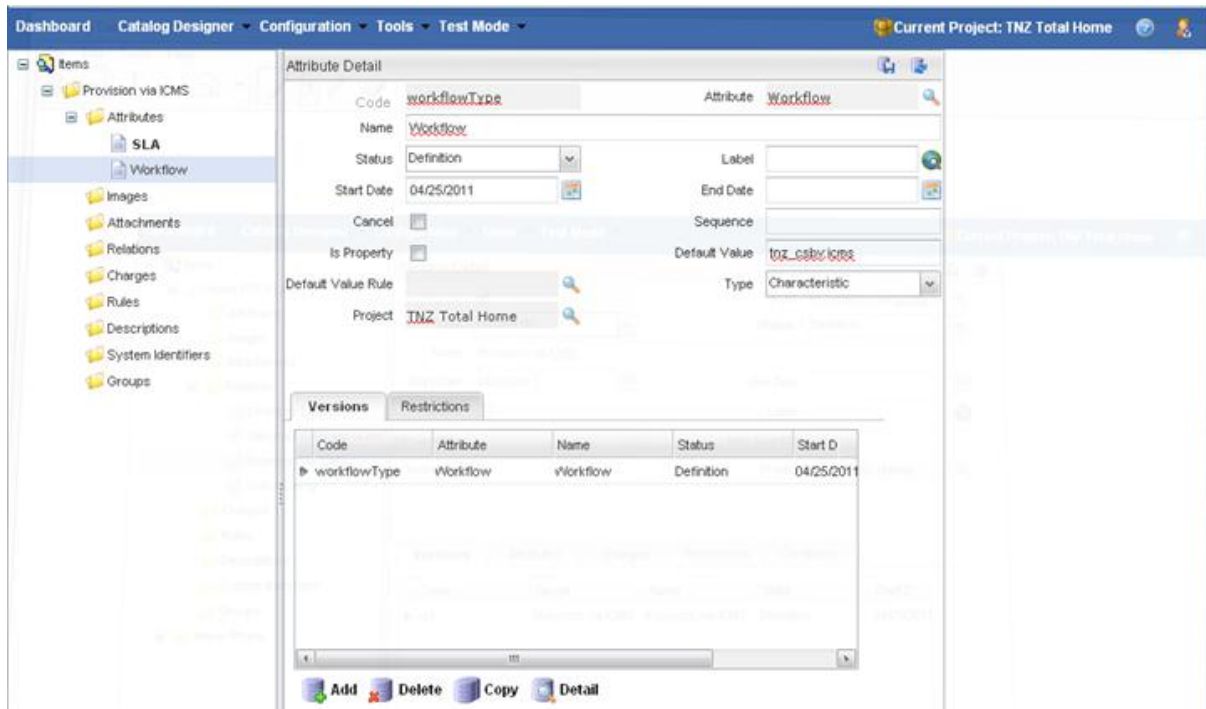
Whether modeled a Contract, SLA, Policy or Workflow, each is modeled as an Item in the catalog, as would be a Product. These items are then linked/associated/related to the Product(s) to which they apply. All capabilities and features of product modeling, apply to these items.

To illustrate, the modeling of the Workflow required to implement a Product is used as an example here:

The linkage between a component at any level of a modeled product, and the workflow required to implement it is defined within the catalog by modeling and associated Items of type Workflow. A Base Workflow item is defined that carries common attributes carried by all Workflow, e.g. the name of the workflow to instantiate, and common parameters such as the SLA to drive jeopardy management within Order Management.



Instances of these workflow are then defined in the catalog. One is defined for each physical workflow modeled in the Ericsson Catalog Manager IDE:

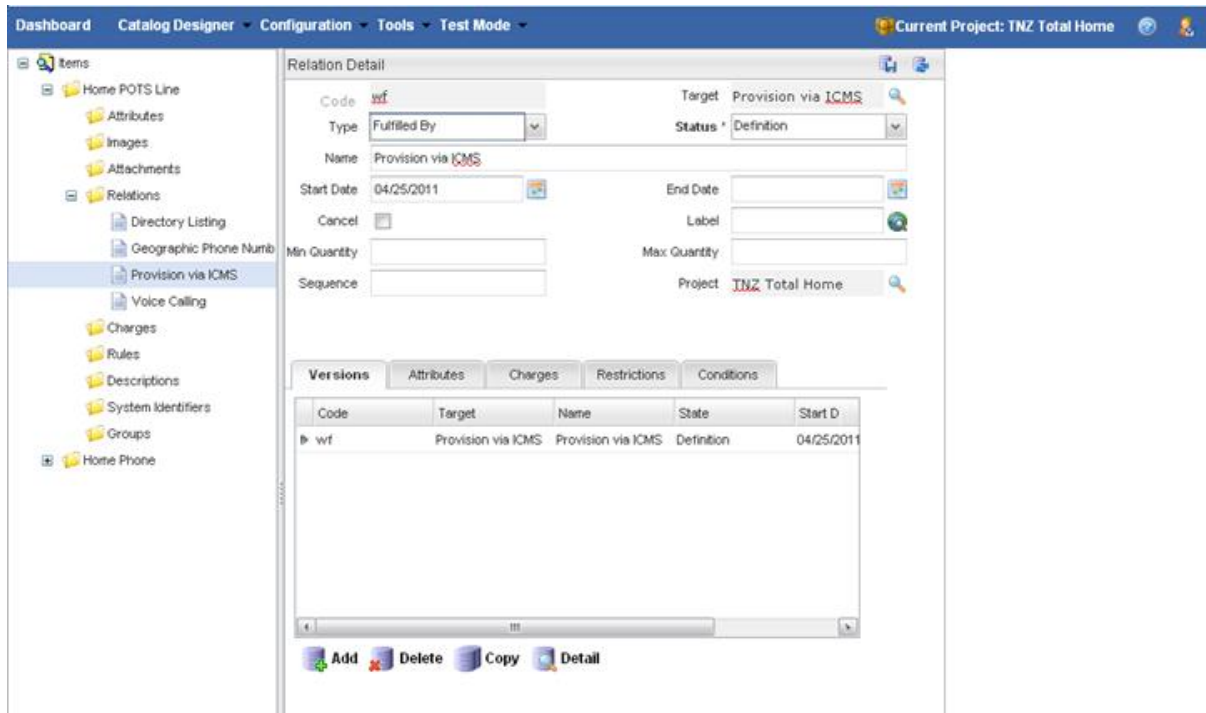


It should be noted that:

- Attribute Restrictions are conditional, permitting workflow variations by any context variable, such as geography or type of client to occur.
- Attribute Restrictions are date effective, permitting the workflow selected to be varied by date



These Workflow items are then associated to the various components of the Product definition. In the following example, the ICMS Workflow is associated with the POTS Line component, which comprises one element of the Total Home bundle:



It should be noted that:

- When associated to a component, the underlying attributes of the Workflow item may be modified to reflect variations from the norm for this particular Product
- Associations are conditional, permitting workflow variations by any context variable, such as geography or type of client to occur.
- Associations are date effective, permitting the workflow selected to be varied by date.
- Multiple workflows could be instantiated for each component.

Once modeled, this information is queried by OM during the Order Decomposition process to drive the workflow selection for the order at any given stage.

The Catalog may be used to model Prerequisites for each workflow by relating the workflow items to the underlying resources and specifying the required state as a relationship attribute.

2.2.5 Mapping to Third-party Systems Naming or Aliasing

Using a billing system as an example, several alternatives exist for the management and association of Billing System Codes, e.g. USOCS, SPOT Codes or Pricing Items, in the catalog.



The mapping of USOCS may be performed in an identical manner to the Workflow Items described in section 3.2.5. In short, a base USOC item is defined that carries common attributes carried by all USOCS, e.g. the code itself. Individual USOCS are then defined within the catalog and associated to the items at the various levels. These relationships and the codes assigned would be conditional and could represent all required business rules. N USOCS could be assigned.

In the event that USOC “groups” are re-used, i.e. 2 or 3 USOCS are always used together, a group of USOCS may be collectively referenced by creating a Master USOC that is itself related to the individual USOCS.

Ericsson recognizes the uniqueness of each third-party system and its coding but notes that all billing systems have their own mechanisms for identifying what was ordered and what needs to be charged. Whether they are SPOT Codes, USOCS, Pricing Items, Charge Codes or any of a number of aliases, the problem is the same. Ericsson Catalog Manager has solved this problem at a number of clients using a variety of billing systems including AMDOCS, Arbor, CSG, ICOMS, Intec, Convergys, CostGuard, Prism, Great Plains, Oracle BRM, and a variety of legacy home grown billing systems.

2.3 Pricing

Charges are defined in the catalog independently from the products themselves. This is done so that they can be re-used across Products. To define a charge or discount, a new Charge Type is created. Each charge type has a recurrence (one-time, monthly, annual, etc.), a currency, effective dates and a set of conditions under which it applies. The conditions may reference any catalog business rule.

Pricing is defined on a line item basis and the consolidated price is defined as the total of the line item prices. There can be any number of charge items for each quoted component, and there is no limit on the total charges across the quote/order. Furthermore, pricing can be calculated based upon the current services held so that volume/threshold based pricing can be applied. In this case access to the client's currently installed/ordered services must be supplied.

Prices for charges are either a flat amount, or one of the pre-set pricing algorithms, such as Stepped or Banded. Other algorithms, including look-ups against price tables defined by Telecom, are implemented via lookups to “Information Models” and Pricing Formulas, which provide a programming like language to defining unique algorithms. Simple look-ups against tables may be defined without IT involvements while the creation of a new script-based algorithm requires IT involvement.

Price changes are defined by creating a new version of the pricing tables linked to the Charge Type, and/or of the Charge Type itself. In either case, the change is date effective.



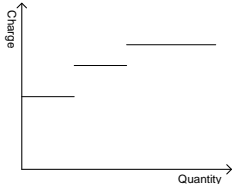
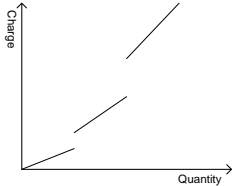
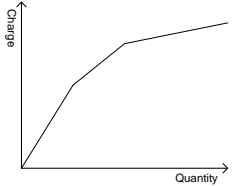
A Price Model can be of default type, or custom type. Default Pricing Model is first discussed, below. Custom Pricing Model is discussed shortly after, in section “Custom Pricing Model”.

2.3.1 Default Pricing Model

In Default Price Model, the Price Model consists of two collections: charges and versions. The *Price Model Charge Type* describes the characteristics of a chargeable item that shall be displayed in an invoice, such as charge recurrence (e.g. one-time, monthly) and method of pricing calculation, and the *Price Model Version* defines a facet of the overall Price Model, such as the applicable date range and eligibility conditions.

Note that the Price Model Charge Type does not contain the amount of the charge itself; the monetary aspect of the charges are tabularized in the Price List of the Price Model Version. This way, the prices of Catalog items can be modified by creating a new Price Model Version (e.g. price hike), rather than modifying the fundamentals of the charge nature in Price Model Charge Type.

In determining the price of a particular item in the Product Catalog, the calculation rule to employ is based on the item’s associated Price Model Charge Type:

Type	Model	Description	Example
Standard Pricing	Not applicable	A static fixed price	Subscription Fee Installation Fee
Standard Pricing, with Formula	Not applicable	Compute using a formula script, based on the Charge and Markup of the matched record in Price List	Cost-based pricing in rural area based on location
Step Pricing – Flat Rate Band		Pricing based on a quantity range; fixed price within range	TV Channels subscription
Step Pricing – Band		Pricing based on a quantity range; fixed rate on all quantity within range	eVouchers for Ring Tones
Step Pricing – Stepped		Pricing based on a quantity range; a defined rate is charged on each incremental range elapsed by the quantity (i.e. “tier rating”)	Pricing based on Data Access Limits



The resulting price is computed by first looking up the monetary entries in the matched record of the Price List of the applicable Price Model Version. These become the input “variables” of the pricing method, from which the resulting price is calculated. To illustrate this procedure, let’s first exemplify with Standard Pricing.

2.3.2 Standard Pricing

Standard Pricing is the most common pricing method – a simple price look-up on the Price List.

For example, the Product “Digital Cable TV service” has a subscription fee of \$50/month and a \$25 Installation fee. A Price Model with two Price Model Charge Types shall be created:

- 1 Digital Cable TV Subscription Fee
 - Monthly charge
 - Standard pricing
- 2 Cable TV Installation Fee
 - One-time charge
 - Standard pricing

The Product shall be associated with both of these Charge Types. The Price Model shall also create a Price Model Version, with a Price List in it with:

Charge Type	Charge	Markup (%)
Digital Cable TV Subscription Fee	50	0
Cable TV Installation Fee	25	0

The Charge in the Price List is the monetary value of Price Model Charge Type, and the Markup is the percentage of desired price increase to the Charge.

Because both Charge Types are Standard Pricing without formula, the resulting price of Charge Type is calculated by the default formula of Charge X (1 + markup/100), resulting in \$50 and \$25 respectively from the Price List lookup. Each of the two Charge Types appears on the Order Invoice as a separate item.

For the recurrence interval of the Charge Types, the Product Catalog does not perform business logic, other than to categorize them in the Order Invoice; a Billing system can integrate and use this information in the Product Catalog for charging and invoicing.

The Standard Pricing can also be accompanied with a tailored formula, which is a JavaScript to compute and return the resulting price. Formulas provide flexibility to satisfy complex pricing requirements such as variable-based pricing, or looking up for information in other systems (e.g. business credit rating).



2.3.3

Step Pricing

The three types of Step Pricing – *Flat Rate Band*, *Band*, and *Stepped* – provide methods to price an item, or item(s) based on quantity.

For example, Component “Internet Hosting” contains an Attribute “Hosting Space” (in megabytes), for which the Component’s yearly price varies according to the Hosting Space that is ordered. For the first 128Mb, it shall be charged 3 cents/Mb. For the next 896Mb, it shall be charged 2 cents/Mb. For the rest up to the 10Gb, it shall be charged 1 cent/Mb, but recently marked up by 20%.

A Price Model with the following Price Model Charge Type shall be created:

- 1 Hosting Space Charge
 - Annual charge
 - Step pricing – Stepped
 - Tailored formula that
 - Assigns quantity: `priceltem.quantity = component.hostingSpace;`
 - Retrieves the price from Price List: `priceltem.getPrice("A");`
 - Returns the result price: `return [priceltem.charge * (1 + priceltem.markup / 100)];`

The Component shall be associated with this Charge Type.

The Price Model shall also create a Price Model Version, with a Price List in it with:

Charge Type	Quantity	Charge	Markup (%)
Hosting Space Charge	128	0.03	0
Hosting Space Charge	1024	0.02	0
Hosting Space Charge	10240	0.01	20

Notice the “quantity” field in the Price List, which defines the upper threshold for a particular step. For instance, if 2Gb of hosting space is ordered, the resulting price shall be $128 \times 0.03 + (1024-128) \times 0.02 + (2046-1024) \times 0.012 = \34.05 from executing the pricing formula.

2.3.4

Custom Pricing Model

In some pricing scenarios, the Price List may not be static on a Charge Type, and it may not only be affected by the quantity ordered. For example, items can be priced differently based on geographic regions, customer loyalty, and/or class of service, etc., which can create a Price List with multiple input variables. Custom Pricing Model enables the creation of Price Lists with additional input attribute(s).



For example, let's say Component "Small Business Internet Service" has an Attribute "CoS" (Class of Service), which can be either "Gold", "Silver" or "Bronze". Internet Gold costs \$99/month, Internet Silver costs \$79/month, Internet Bronze costs \$59/month.

A Custom Price Model can be created with Attribute CoS associated. All Price Lists in all Price Model Versions of this Price Model shall have CoS as a column.

With a Price Model Charge Type setup as:

- 1 Small Business Internet Subscription Fee
- 2 Monthly charge
- 3 Standard pricing
- 4 Tailored formula that
 - Assigns component CoS to pricing: `priceltem.cos = component.cos;`
 - Retrieves the price from Price List: `priceltem.getPrice("M");`
 - Returns the result price: `return [priceltem.charge];`

The Price List can then be populated as follows:

Charge Type	CoS	Charge
Small Business Internet Subscription Fee	Gold	99
Small Business Internet Subscription Fee	Silver	79
Small Business Internet Subscription Fee	Bronze	59

2.4 Discounting And Promotions

Discounts are defined in exactly the same manner as charges. In fact, discounts are modeled as charges but have the "discount" flag checked.

Discount functionality includes the ability to define a discount amount, percentage or formula, to define the eligibility rules, and to define which individual charge(s) the discount applies to. Supported options include 5% off all services (excluding regulatory charges), \$10 off a single service's subscription charge, and a waived installation charge.

Discounts may be defined to be "display purposes only". Such discounts are employed to communicate discounts and promotions that do not affect the quoted price of the services. For example, they may be used to indicate promotions such as "free calling on your birthday", "eligible for volume discounts if monthly spending exceeds \$250", etc.



2.5 Taxation

Ericsson Catalog Manager provides a taxation engine that provides full taxation functionality. Taxation services in markets where numerous taxes apply are typically handled via an interface to a third-party taxation system. Generally in markets where state and federal taxes are in play, Ericsson Catalog Manager's taxation services may be effectively deployed.

Ericsson Catalog Manager's taxation facilities include:

- The ability to define a "model" that defines the sequence of tax application and to define the base amount upon which each tax is calculated (gross, inclusive of prior taxes). This model concept enables variation in tax application rules across jurisdictions, such as the provinces of Ontario and Quebec in Canada.
- Taxes are defined independently of individual charges and by default apply to all charges. Eligibility rules limit their application, such as to a particular province, or to a particular class of charges. These rules are also employed to govern tax exemptions.
- There is no limit to the number of taxes that may be applied.
- Taxes are calculated at the line item level.
- Taxes are date effective and versioned (as are all prices and discounts)

2.6 Product Catalog and Service Catalog

The Ericsson Catalog Manager can be used to model Products and Services through the use of "items". Products, Services and Resources are all "Items" within the Ericsson Catalog Manager. And are differentiated through an SID Entity Type, i.e. Product Offering, CFS, RFS, Logical Resource, Physical Resource, etc. These items are then related to each other using the composition relationship:

- A relationship type of "Comprised Of" is used to relate the parent Item to its constituent components, e.g. the Product to the CFSs, and RFSs.
- Conditional relationships are supported, e.g. Product1 consists of CFS1 when in New Zealand (and CFS2 in Australia).
- Group Selection is supported, e.g. Product1 consists of 1 item in the Group, that itself consists of CFS1 through 5. At Least, At Most and Exactly are supported.

Decomposition of a Product into Services and Resources, is possible then by asking the catalog for all Items of the "Comprised Of" relationship for a given Product, in a given Context (e.g. location=Melbourne, digital serving area, etc.). The Catalog Manager will perform and return the fully decomposed product, i.e. if a constituent service is itself decomposed into services and resources, the Catalog Manager will return that decomposition as well, to any depth.

The following sections provide the recommended mapping for selected entities.



2.6.1 Product Offering



Item Type: Product Offering

Relationships:

- Comprised of 1:N Product Specifications
- Restricted by 0:N availability rules
- Restricted by 0:N eligibility rules
- Validated by 0:N validation rules

Characteristics:

- May specify new characteristics (not typically)
- Specifies any Characteristic Values required for that specific offer for Characteristics either defined locally or inherited through the composite relationships

2.6.2



Item Type: Product Specification

Relationships:

- Comprised of 0:N CFS Specifications
- Comprised of 0:N Physical Resource Specifications
- Restricted by 0:N availability rules
- Restricted by 0:N eligibility rules
- Validated by 0:N validation rules

Characteristics:

- May specify new characteristics
- Specifies any Characteristic Values required for that specific offer for Characteristics either defined locally or inherited through the composite relationships



2.6.3 Customer Facing Service Specification



Item Type: CFS Specification

Base Type:

- Service Specification

Relationships:

- Comprised of 0:N RFS Specifications
- Restricted by 0:N availability rules
- Restricted by 0:N eligibility rules
- Validated by 0:N validation rules

Characteristics:

- May specify new characteristics
- Specifies any Characteristic Values required for that specific offer for Characteristics either defined locally or inherited through the composite relationships



2.6.4 Resource Facing Service Specification



Item Type: RFS Specification

Base Type:

- Service Specification

Relationships:

- Comprised of 0:N Resource Specifications
- Restricted by 0:N availability rules
- Restricted by 0:N eligibility rules
- Validated by 0:N validation rules

Characteristics:

- May specify new characteristics
- Specifies any Characteristic Values required for that specific offer for Characteristics either defined locally or inherited through the composite relationships



3 Application Virtual Machine (AVM)

The Ericsson Catalog Manager AVM is the runtime framework of EricssonCatalog Manager applications. Starting the AVM requires that you have your project's metadata open and saved in Service Designer. Service Designer also allows you to stop the AVM at any time. You can also start the AVM when you are debugging your project metadata.

The following Web-based applications comprise the Ericsson Catalog Manager AVM:

- System Administration Application, which provides detailed, real-time information about running the administration application, including the status of all AVM nodes, and defining users, groups, and privileges.
- Configuration Tool, which enables the runtime configuration of applications and allows you to configure across application nodes when applications are deployed in a J2EE cluster setting.

Applications that you have created in your project metadata are also available when the AVM is running.

3.1 AVM Cache

The cache is a generic mechanism in the AVM that stores in memory data, which can be refreshed or recreated from disk (database or files).

You can configure the AVM cache by using the following parameters in the Configuration Tool:

- **Capacity**, which defines how much memory in MB the cache can take.
- **Element size**, which is defined in KB and is automatically set by the Configuration Tool.
- **Refresh schedule**, which defines one or more points in a 24-hour day when the whole cache content will be dropped. The cache will then rebuild itself dynamically as new requests come in.
- **Automatic reload**, which allows you to specify the percentage of an element that is reloaded.
- **Interval (hours)**, which refers to the time in hours in which the cache writes its current content to the database.



3.2 AVM Cache Types

AVM cache types can be configured from the Ericsson Catalog Manager Configuration Tool. The following table describes the names and a short description of each cache role.

Name	Description
CodeTable Cache	Contains code tables used mainly by the UI server.
Document Cache	Consists of read-only copies of documents.
ExternalOrderId Cache	Contains the relationship between the external order ID and order ID. This is companion cache for the order cache.
FinderResult Cache	Consists of data from the large finder result sets.
Image Cache	Contains images and is used only by the UI server.
InterfaceData Cache	Consists of interface data.
Order Cache	Contains orders.
Reference Cache	Consists of references and is used predominantly by the UI server.
Resource Cache	Contains different type of resources (mainly text), such as XSLT specs, browser JavaScript programs, etc.
Script Cache	Consists of metadata and catalog scripts.
Translation Cache	Contains the language translations of translation data types.



3.3 Certified Environments

3.3.1 Third-party Software

As of December 15th 2011, Ericsson Catalog Manager has been certified to support the following technologies:

Operating Systems		J2EE Application Servers		
		JBoss 5.1	Oracle WebLogic 10.3*	IBM WebSphere 7.0.0.9 Note
		Sun JDK 1.6.0.16	Sun JDK 1.6.0.5+	WebSphere Embedded JDK
Windows	Server 2008 Windows XP Pro SP3	☑		
AIX	6.1			☑
HP-UX	11.31 (11 v3)		☑	☑
Sun Solaris	10	☑	☑	☑
Linux	Red Hat 5	☑	☑	☑
Database	Oracle 10g R2 & Oracle 11g R2			

Ericsson Catalog Manager's software supports the following browsers:

- Firefox 3,4,5
- Microsoft Internet Explorer 6, 7, 8, 9
- Google Chrome (latest stable)
- Apple Safari 5

Ericsson Catalog Manager endeavors to support Grade A browsers as defined by YUI's Graded Browser Support. The current list can be found at <http://yuilib.com/yui/docs/tutorials/gbs/>.

3.3.2 Client Hardware

The specifications for PCs are relevant for a single application use. If PCs are to be used for multiple applications and/or additional functions, then the minimal configurations will need to be increased.

Application	Unit	Minimum Specification	Recommended Specification
End Users of Application	Any	800 MHz Pentium III 256 MB RAM	Pentium P4 2.4 Ghz 512 MB RAM
IT users running IDE	PC – Standard Desktop MS-Windows 7 & XP	2 GHz CPU 1 GB RAM 2 GB or larger disk	Dual 2 GHz CPU 4 GB RAM 4 GB or larger disk



3.4 High Availability Deployment Architecture

The Ericsson Catalog Manager product is built on top of leading industry infrastructure products – web servers, application servers, database servers and EAI middleware. As such, high availability requirements for the Ericsson Catalog Manager solution are largely handled via the commercial application servers and database servers that support the product. As an example, Ericsson Catalog Manager has a large tier 1 customer that achieves high availability implementation using a BEA clustered environment with a hardware-based load balancer to support application server clustering.

3.4.1 Ericsson Catalog Manager High Availability Design Characteristics

A number of characteristics of the Ericsson Catalog Manager design contribute to maximizing the availability of the Ericsson Catalog Manager solution.

3.4.1.1 Application Servers

- Store no business data - all business and system configuration information is stored in the database.
- Can be configured to handle user interactions, API interactions and/or workflow.
- Each instance of a workflow server (a.k.a. process engine) may be allocated only a portion of the workflow load by type of process.
- Each instance of a workflow server is assigned an ID within a cluster of set size. Work is automatically distributed within the cluster using a hash scheme that ensures all activities for a process instance is handled by the same workflow server.

3.4.1.2 High-Availability Configuration

- Webserver and/or hardware based load balancers. Perform load-balance and fail-over between applications servers.
- N-Application servers managed as an application server based cluster
 - Each user is serviced by one application server.
 - Each process is serviced by one workflow engine.
- Hardware based load balancers managing database transactions
- High availability database. E.g. Veritas clustered server solution for Oracle.

3.4.2 Monitoring

Each server maintains a heartbeat interaction with the database, and via JMX, allowing for third-party tools to monitor the health of the server. The JMX based interface exposes a rich set of statistics that can be used to determine not only running/not running state, but also the performance, memory utilization, and thread utilization.



3.4.3 Impact of Failed Components

3.4.3.1 Application Server – User Load

Sticky sessions need to be configured within the load balancer/web server as there is no user session swapping between application servers. As a result, in the event of a failed application server, the users being served by the server will lose their connection and must re-login. All business data from the prior session will be saved/retained from the last successful interaction. Any in-progress transactions are rolled-back at the database level. The web-servers/load balancers will automatically handle the re-direct to another application server.

3.4.3.2 Application Server – Workflow Server / Process Engine

Each server will attempt to remain operational in the event of a problem:

- On loss of database connection, server will immediately stop processing and attempt to reconnect periodically until the connection is restored or the system is restarted.
- On a low memory condition, alternate memory management routines are triggered, including management of system cache sizes and retention periods, and ultimately, suspending the processing of new workflow processes.
- On a non-recoverable failure, a clean-and-restart procedure is followed that will automatically reset the workflow engine to an initial state. The restart is logged so that it can be later diagnosed. No work is lost and work continues seamlessly.

Hot stand-by nodes can be configured so that a permanent failure of one “working” node, such as due to a hardware failure, can be automatically recovered by the stand-by node. This ensure an automated, seamless and immediate restoral of service. In the event of a permanent failure, any in-progress transactions are rolled-back at the database level. Upon restart, workflow is transparently resumed from the last uncommitted activity within each workflow instance.

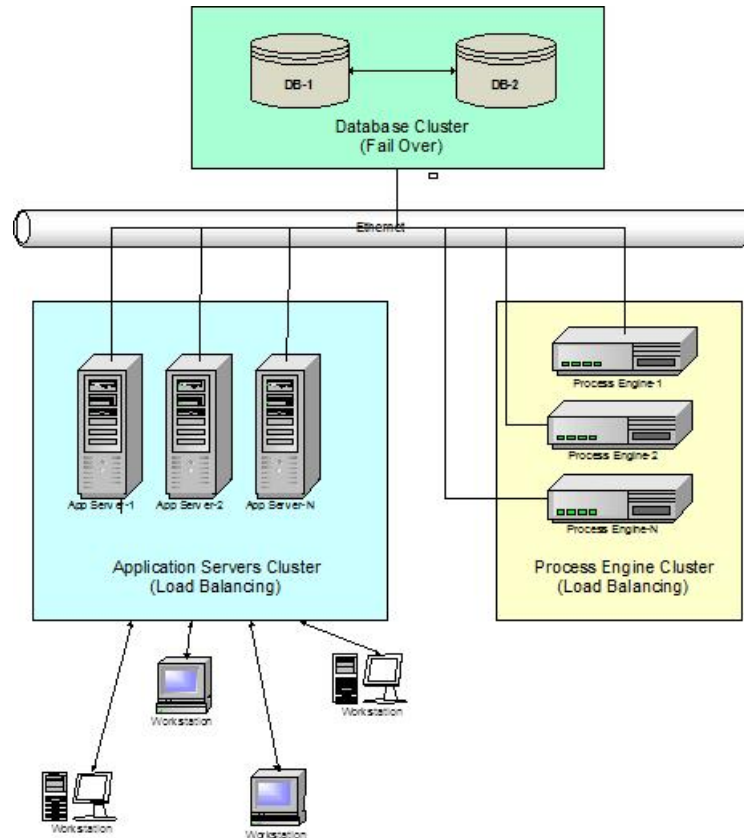
3.4.3.3 Database Server

On loss of a database, servers will immediately stop processing and attempt to reconnect periodically until the connection is restored or the system is restarted.



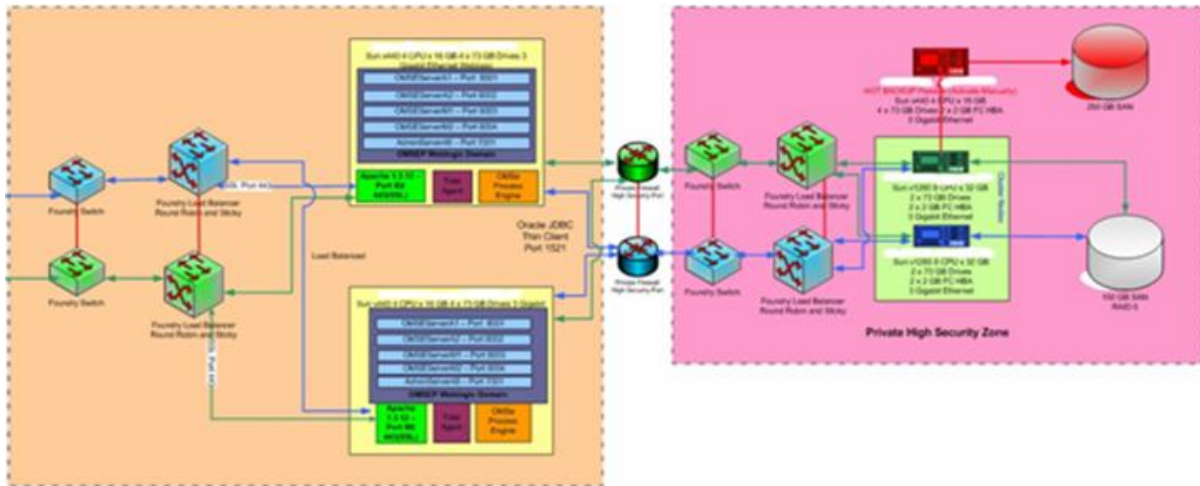
3.4.4 Typical High Availability Design

The following diagram depicts a typical high availability design in which the user and workflow load has been split into two clusters of application servers in order to isolate the impact of bursts in traffic/workload from one type of activity on the other.



Ericsson Catalog Manager's highest volume deployment handles a peak volume of 40 orders/second, or 120,000 orders / hour. The following deployment architecture is for a client solution that supports over 20,000 orders per day with a community of over 10,000 users. This deployment is characterized by:

- Hardware based load balancer and SSL accelerator distributing user and API load to the application servers
- BEA WebLogic clustered application servers with selected nodes dedicated to user traffic, and others to API traffic
- Hardware based load balancer and SSL accelerator distributing load to database layer
- Active / Passive Clustered database solution (active failover)
 - Manual failover to alternate Database Server
 - SAN based storage

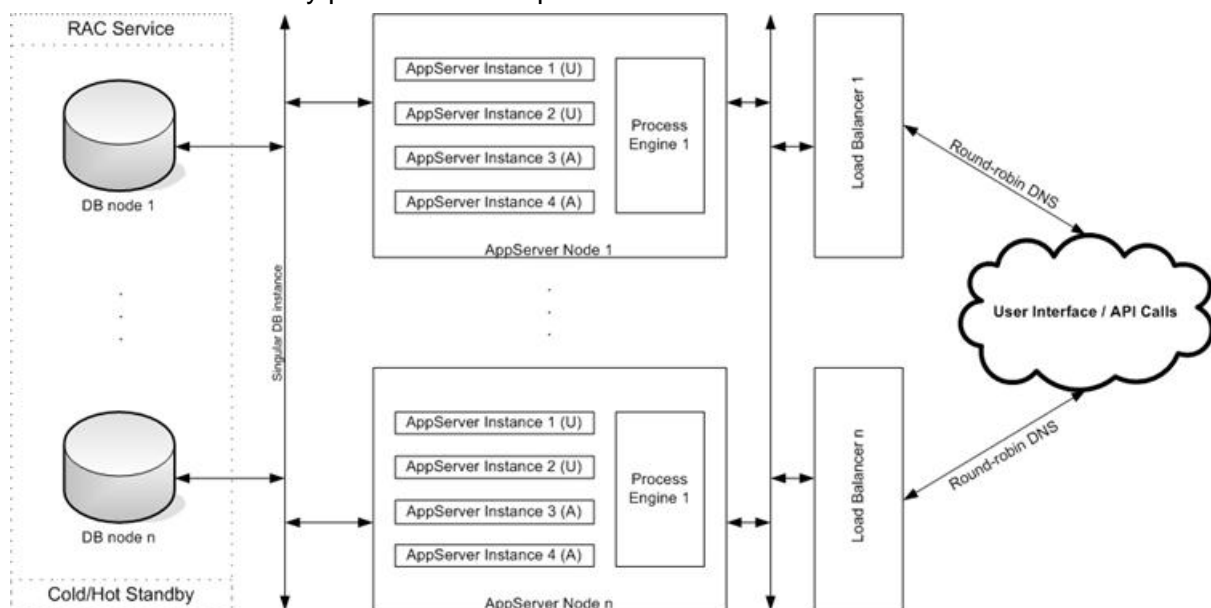


3.5 Deployment Sizing

In general, performance sizing is largely dependent on the underlying metadata which implements an application as well as the complexity of the business processes which are being modeled. The ideal solution to sizing hardware accurately – relies upon designing metadata and conducting initial performance testing prior to the final assignment of hardware.

3.5.1 Sample Configuration

It is suggested that a single App Server Node and a single DB Server Node be implemented based on the anticipated system load expected in terms of transactions per second. With this configuration in place, a series of performance tests can be conducted to ensure that the system is responding as designed. Furthermore, depending upon the performance results, one can then determine what additional hardware/software refinements are needed to satisfy performance requirements.





As one can see, each App Server Node has been configured to have 4 Application Server instances. This is primarily to handle the UI load of users as well API based Web Service requests to the Ericsson Catalog Manager Framework from external systems. Additional Application Server instances can be added as WEB based traffic increases, however; care should be taken to ensure that adding instances does not degrade the overall performance of the system. In such a scenario additional hardware App Server Nodes will have to be added to the server farm.

Each App Server Node has also been configured to have one instance of the Process Engine. Process Engine instances implement the workflow engine of the Ericsson Catalog Manager Product Suite and are responsible for moving orders along business process models. There is no direct interaction between users of the system and the Process Engines, moreover; the Process Engine spends most of its time in performing database operations. Hence, depending upon the complexity of the process models within a metadata; it may be necessary to move Process Engine instances onto separate independent hardware for greater throughput.

3.5.2 Generic Hardware Sizing

Ericsson Catalog Manager will provide guidelines regarding the hardware that is required to meet the proposed solution. Factors affecting the proposed hardware configuration include:

- Real-time demand for order creation (number of orders)
- Number of simultaneous users
- Database access requirements
- Number of processes and associated activities
- High-availability requirements

Specifically, the following metrics are collected in preparing the hardware sizing recommendation:

- Concurrent users
- Orders per month
- Business days per week
- Hours per work day
- Transactions per order
- # of busy hours per day
- Expected order life cycle duration (by type of order)
- Transactions per hour within a busy hour
- Transactions per second within a busy hour



Based upon Ericsson Catalog Manager testing and customer experience, the following table provides a “rule-of-thumb” for determining the recommended number of servers, and their technical characteristics. This table is based on the number of orders per day, but as stated earlier a number of other drivers must be considered as well.

Scenario	Orders per Day	Number of Servers	Server Function	CPU (cores)	Clock	RAM	Hard Disk
Low	<=500	1	Application/PE	8	1 to 2 Ghz	8 GB	60 GB
		1	Database	8	1 to 2 Ghz	8 GB	80 GB
Medium	500 to 5000	2	Application/PE	8	1 to 2 Ghz	8 GB	60 GB
		1	Database	8	1 to 2 Ghz	16 GB	100 GB
High	5000 to 80,000	3	Application/PE	8	1 to 2 Ghz	8 GB	60 GB
		2	Database	16	1 to 2 Ghz	32 GB	250 GB

Ericsson Catalog Manager recommends the required processing power, memory requirements and disk space of the underlying hardware but does not recommend a specific hardware as the final selection will be based upon many additional factors such as the availability of existing servers, pricing, and standard operating platforms.

3.5.3 Additional Comments

The Web Server layer used in some architectures is not required from a Ericsson Catalog Manager perspective.

Ericsson Catalog Manager recommends a hardware based load balancer rather than a software based load balancer from BEA or Apache. BEA has been offering software based load balancing for some time while Tomcat has recently added this capability. While both have solutions, all of Ericsson Catalog Manager’s customers to-date have chosen a hardware accelerated load balancer solution.

The Application Server and Process Engines do not store any business data on local (or remote) storage. All business and system configuration data is stored in the Oracle database. The local server must, however, have sufficient disk space (we have recommended 80GB or greater) to house:

- The application itself
- All third-party software, e.g. BEA WebLogic
- Application logs

Ericsson Catalog Manager is not aware of any compatibility issues between its software and VMWare or running under a virtualized machine. Ericsson Catalog Manager believes the approach is viable but has no experience with customers running its software under VMWare or VM’s. Due to the size of the servers recommended, Ericsson Catalog Manager believes that stand-alone servers may be more appropriate.



3.5.4 SPEC Performance Benchmarks

Ericsson Catalog Manager recommends the required processing power, memory requirements and disk space of the underlying hardware but does not recommend a specific hardware as the final selection will be based upon many additional factors such as the availability of existing servers, pricing, and standard operating platforms.

SPEC performance benchmarks are used to scale the required specifications to the desired target platform. Typically we compare the commodity (Intel) architecture employed within Ericsson Catalog Manager benchmark tests to the client's desired platform and scale up/down as required. An extract of dated SPEC benchmarks from 2005 follow as an example only:

Tester Name	System Name	System Web Server Software	Script	CPU				Result
				Cores	Chips	Cores per chip	HW Multi-threading	
Dell	PowerEdge 1950	Rock Web Server v1.3.3 (x86-64), Apache Tomcat 5.5.17	JSP	4	2	2	No	9808
Dell	PowerEdge 2850	Zeus Web Server 4.2r4 (x86-64), Apache Tomcat 5.5.9	JSP	4	2	2	Yes	4850
Dell	PowerEdge 2950	Rock Web Server v1.4.0 (x86_64), Rock JSP/Servlet Container v1.2.0 (x86_64)	JSP	4	2	2	No	14495
Dell	PowerEdge 750	Zeus Web Server 4.3r1 (32 bit), Apache Tomcat 5.5.9	JSP	1	1	1	Yes	848
Sun Microsystems Inc.	Sun Fire T2000	Sun Java[TM] System Web Server 6.1 SP5 64-bit,	JSP	8	1	8	No	14001
Sun Microsystems Inc.	Sun Fire T2000	Sun Java[TM] System Web Server 6.1 SP5 (64 bit),	JSP	8	1	8	No	16407
Sun Microsystems Inc.	Sun SPARC Enterprise T2000	Sun Java[TM] System Web Server 6.1 SP5 (64 bit),	JSP	8	1	8	No	16407
Sun Microsystems Inc.	Sun SPARC Enterprise T5220	Sun Java[TM] System Web Server 7.0 Update 2,	JSP	8	1	8	No	37001

Source: <http://www.spec.org/web2005/results/web2005.html>

3.5.5 Sample Recommendations

The following are extracts from a sample hardware sizing performed for a client and provide additional insight into the factors to be considered.

3.5.5.1 Commodity Hardware Specifications

Ericsson Catalog Manager recommends the following hardware specifications:

- (2) Load Balancers. Most of our customers had success with the F5 brand of load balancers.
- (2) A robust database server with at least 16 GB RAM each (or better) connected via SAN to a sufficiently sized data store with enough bandwidth exceeding the maximum anticipated throughput in terms of



transactions per seconds. We estimate that 300GB should be sufficient to accommodate a couple of years of data. Data archiving and backup strategy is left to the internal corporate IT policies and procedures of the client. The IOPS figure is estimated to be around 232 KB/s. We use the following rationale to arrive to our IOPS figure:

$((4 \text{ orders/sec} * 6 \text{ KB}) + (50 \text{ transactions/sec} * 3 \text{ KB})) * 1.33$
overhead = 231.42 KB / sec.

- For each database system a typical real-life example could be a Dell PowerEdge 2900 III Server configured with (2) Quad-Core Intel Xeon Processors 5400 series (3.16GHz) starting with 16GB of RAM up to a maximum of 48GB RAM per system.
- (4) dedicated application servers systems configured with at least 4GB RAM each (or better). 8GB would be more appropriate, considering that in our rule-of-thumb generic table, we specify under the high volume scenario that the Application Server should have 4GB and the Process Engine also should have 4GB and be separate. However the load is distributed across four distinct systems. However, Ericsson Catalog Manager recommends starting with 8GB per application server system, in order to allow scaling up to multiple (logical) instances of the BEA AppServer if the need arises over time.
- A typical real-life example could be a Dell PowerEdge 2950 III Server configured with (2) Quad-Core Intel Xeon Processors 5400 series (3.16GHz) starting with 8 GB of RAM up to a maximum of 32GB RAM per system.

Since there is a great variety of CPU architectures, with vastly differing performance and costs, the generic recommendation is to ensure that each CPU has enough processing power to sustain the anticipated load. With today's modern processors, it is not uncommon to find 4 core processors that exceed the required performance requirements at commodity prices. However, the better the CPU, the better the overall system performance, thus CPU selection becomes an issue out of the scope of this document, better governed by available budgets and / or partnering with various hardware and / or system integrators.

But as a suggestion, going with the high-end Intel products from solid vendors such as Dell, HP or IBM may not be a bad idea. Alternatively, Sun offers very high performance products using the Sun SPARC or AMD architecture.

Each CPU, regardless of architecture, is usually complemented with 4GB of RAM for large volume of transactions. Therefore a typical (4) CPU system, would be augmented with 16GB of RAM.

3.5.6 Recommendations on Development, Testing, and Production Environments

There should be at a minimum a Development environment, a Performance Testing/QA/Staging environment, and finally the Production environment.



The production environment is depicted in the recommendations provided. The other environments would ideally mimic the production environment, but budgetary constraints may prevent this scenario as being realistic. However, since the non-production environments won't be subjected to a continuous high volume of real transactions, they may be scaled down with the exception perhaps of the Performance Testing environment.

3.6 Performance Management and Monitoring

Ericsson Catalog Manager includes a Performance Management component that permits monitoring the system, including both system and user defined metrics and events. Included are facilities to publish these statistics to third-party systems, including a JMX based interface. Native adaptors are provided for the NT Performance Monitor and HP Open View. A generic output stream is provided and can be used by most monitoring solutions, including Tivoli.

3.6.1 JMX Based Management and Monitoring

Ericsson Catalog Manager supports JMX as a monitoring and administration protocol. Through the JMX interface, CSP's would be able to perform all monitoring functions through JConsole, or any other monitoring tool that supports JMX. Adaptors also exist for HP OpenView and the NT performance monitor. File based output is also available for monitoring tools that do not support JMX.

Item monitored include the status of the workflow engines (are they actually running), memory utilization, health of system caches (size, hit rates, overflow events), number of processes, number in error, and detailed status of each worker thread. A screen capture of a portion of the statistics available through JMX, are captured in the following screen capture.



The screenshot shows the J2SE 5.0 Monitoring & Management Console. The left pane displays a tree of MBeans under the package `com.conceptwave.AVM`. The right pane shows the 'Attributes' tab for the selected MBean, displaying a list of attributes and their values.

Name	Value
ActivityCommandRatePerMin	13
ActivityQueueSize	0
ActivityRatePerMin	21
AsymmetricClusterPE	false
BusyWorkingThreadsCount	0
CategoryPrinting	false
CwPortNumber	9900
DelayedActivityQueueSize	6
FreeMemoryLimitMB	100
FreeMemoryMB	341
GlobalProcessSerialization	false
LowMemoryEventCount	0
NumberOfFinishedProcesses	0
NumberOfGlobalProcesses	13
NumberOfProcesses	13
NumberOfProcessesInError	0
NumberOfRestartsOfGlobalProcesses	13
NumberOfResumedProcesses	13
NumberOfStartedProcesses	13
NumberOfSuccessfullyCompletedProcesses	0
NumberOfTopLevelUserProcesses	0
NumberOfUserProcesses	0
NumberOfWorkingThreads	25
OrderSynchronization	false
PEId	0
PEStatus	Running
ParticipantQueueSize	19
PerformedActivityCount	21
PerformedActivityCommandsCount	13
ProcessGroupSynchronization	false
ProcessStartRatePerMin	13
ProcessesLimit	172000
RequestQueueSize	0
Speed	0

The statistics provided are provided in real time. Tools such as JConsole may then provide real-time graphical dashboards.

3.6.2 User Defined Metric Definition

The screenshot shows the 'ConceptWave Configuration' dialog box. The 'Enable performance counters' checkbox is checked. The 'Collecting frequency (seconds)' is set to 1. The 'Output type' is set to 'Standard output'. There are four 'User counter' fields, each with a text input and a dropdown menu for aggregation.

User counter	Value	Aggregation
User counter 1	User-1	Total
User counter 2	User-2	Total
User counter 3	User-3	Total
User counter 4	User-4	Total

Metrics are collected based upon defined intervals and aggregated over configurable time periods. The measurement intervals for the metrics are configurable. The resulting measurements can be used to perform trending, either within third-party monitoring tools, or using the reporting capabilities described in the Order Analytics (OA) module section.

3.6.3 Fault Management and Monitoring

The Ericsson Catalog Manager application can be monitored on a regular basis using the Systems Administration Tool. As an example, staff can access the following screen to ensure that the Framework and Process Engine applications are running.



General Framework Information		
Framework		
Config File	Version	
CwfConfig.xml(embedded)	3.0.43	
Session Timeout (min)	Test Mode	
90	<input type="checkbox"/>	
Metadata		
Metadata Host	Metadata Label	Metadata Name
EM	Created by 'JacobFussek'	EM: Mar. 15 e
Metadata Timestamp	Metadata Version Number	
Mon Mar 15 14:43:19 EST 2004	34	
License		
Licensed To	Expires	
ConceptWave internal	30.04.04	
Users	Process Engines	Evaluation
50	2	<input type="checkbox"/>
Order Entry	Process Manager	Designer
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Process Designer		
<input checked="" type="checkbox"/>		
Servers		
Running	Login Users	Guest Users
1	0	0
Weekly Error Event Count	Max Login Users	Max Guest Users
245	0	0

Additionally, alerts can be triggered by:

- Business flow alert activity - As part of the process flow, an alert activity can issue an alert to indicate and/or escalate a business error.
- Process abnormal termination - This alert is dispatched when a process instance terminates prior to completing all process activities. The causes for process abnormal termination include such conditions as JavaScript errors, failure to handle process exceptions, and database errors.
- Process engine abnormal termination - This alert is dispatched when the Process Management system detects a Process Engine failure. Process Engine abnormal termination can be caused by server shutdown or by Java VM (virtual machine) shutdown.

Both real and non-real-time monitoring is supported. The alerts can be delivered to an email, pager, external system, or work queue.

3.6.4 Inventory Management

Ericsson Catalog Manager software is deployed on top of a J2EE application server and hardware managed and monitored by user/customer, within its data center. Inventory management of these core components is supported within the clients existing processes.

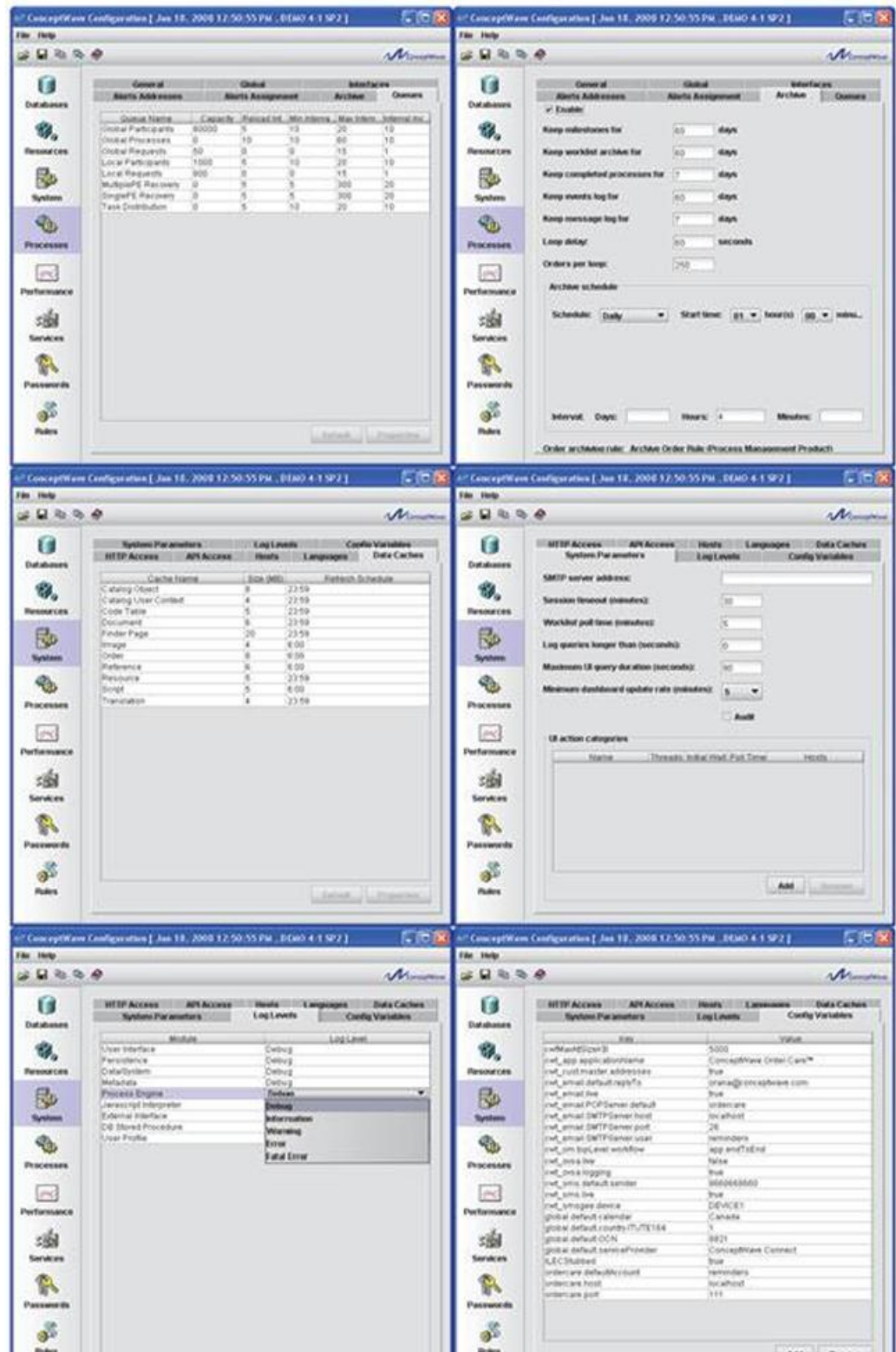
3.6.5 Server Configuration Management

Similarly, the core J2EE application server, operating system and hardware configuration is standard and typically performed by customer itself. Ericsson Catalog Manager components are configured through the Configuration tool, which provides control over the system behavior, resource profile and performance. Configurable items include:



- Logging levels
- Archiving: frequency, duration and rules
- Caches: size, refresh intervals
- Workflow engines: number; threads; roles
- Background processes: number, servers
- Database: locations, schemas and passwords
- Interfaces: IP addresses, ports, queues, binding settings, passwords, etc.
- HTTP: port filtering, HTTPS settings

Sample screen prints follow on the next page:





3.7 Auditing and Event Logs

As order creation is started, the Ericsson Catalog Manager solution audits and tracks all aspects of the order lifecycle. From the users who touch the orders to the milestones that are completed, timestamps of all activities are stored in our database for immediate retrieval. Auditing of changes is provided at the field level for those fields selected, and can be turned on/off via scripts. Audit trails are also provided on each work queue as tasks are taken/returned/reassigned/completed. Logging is provided on all errors to a database table. The severity level of errors to be logged is configured. Interface logging is also available on an individual interface level.

In addition, the Ericsson Catalog Manager Systems Administration application includes an Event Log that displays a list of messages logged by the application. An example of the Event Log is shown below.

The screenshot shows the 'Event Log' window with a search bar at the top and a table of log entries below. The table has columns for ID, Date, Module, Severity, Code, User ID, and Description. The log entries include information about user login, system startup, and missing images.

ID	Date	Module	Severity	Code	User ID	Description
2791	16.03.04 13:45:28	User Interface	Information	2	<login>	Session "HttpSession[<login>(24126FFF0DA720D1
2790	16.03.04 13:45:26	Data/System	Information	9005	<cwfsysinit>	16M memory used at startup
2789	16.03.04 13:45:25	Data/System	Information	9109	<cwfsysinit>	ConceptWave system started
2788	16.03.04 13:45:17	Data/System	Warning	1	<cwfsysinit>	Missing image: "/report/work_item_sum.gif".
2787	16.03.04 13:45:17	Data/System	Warning	1	<cwfsysinit>	Missing image: "/report/work_item_stat.gif".
2786	16.03.04 13:45:17	Data/System	Warning	1	<cwfsysinit>	Missing image: "/report/multi_center_wl.gif".
2785	16.03.04 13:45:17	Data/System	Warning	1	<cwfsysinit>	Missing image: "/report/multi_center_wl.gif".

The event log finder can send manually selected log entries as XML to a default or user specified e-mail address. Filtering of messages stored in the event log can be achieved based on userID or context parameter (for example user name, context detail, etc.). A set of system configuration parameters including but not limited to debug level and event log filtering criteria can be changed on run time without restarting the system.

Event logs will continue to be collected and stored as long as enough database storage is available to support it. The archiving function of the Ericsson Catalog Manager solution allows for retention periods to be configured for event logs.

A configurable archiving facility operates on all logs (Audit, Message and Event). The archiving function will automatically purge aged logs past the expiry time of the retention period.



The following illustrates the data captured for selected logs:

Audit Trail

Column Name	Data Type	Nullable
ORDERID	VARCHAR2(16 BYTE)	No
DOCID	VARCHAR2(16 BYTE)	No
FIELDNAME	VARCHAR2(128 BYTE)	No
DOCTYPE	NUMBER(9,0)	No
ORIGVALUE	VARCHAR2(256 BYTE)	Yes
CHANGEDVALUE	VARCHAR2(256 BYTE)	Yes
LASTCHANGED	DATE	Yes

Change Log

Column Name	Data Type	Nullable
RECNO	VARCHAR2(16 BYTE)	No
DOCID	VARCHAR2(256 BYTE)	No
DOCMETADATATYPE	NUMBER(9,0)	No
TOPORDERID	NUMBER(16,0)	Yes
TOPORDERMETADATATYPE	NUMBER(9,0)	Yes
UPDATEDBY	VARCHAR2(64 BYTE)	No
LASTUPDATEDDATE	DATE	No
OPERATIONTYPE	VARCHAR2(3 BYTE)	No

Logged Users

Column Name	Data Type	Nullable
USERID	VARCHAR2(64 BYTE)	No
SERVERID	NUMBER(9,0)	No
STATIONIPADDR	VARCHAR2(15 BYTE)	No
ACTIVE	NUMBER(3,0)	Yes
LOGEDIN	DATE	Yes

Interface Log

Column Name	Data Type	Nullable
MSGID	NUMBER(16,0)	No
VMID	NUMBER(9,0)	No
INTER_TYPE	NUMBER(9,0)	No
OPERATION	VARCHAR2(256 BYTE)	No
USER_ID	VARCHAR2(64 BYTE)	No
USER_DATA1	VARCHAR2(64 BYTE)	Yes
USER_DATA2	VARCHAR2(64 BYTE)	Yes
USER_DATA3	VARCHAR2(64 BYTE)	Yes
CREATION_TIME	DATE	No
SEND_DATA	BLOB	Yes
SEND_TIME	DATE	Yes
RECEIVE_DATA	BLOB	Yes
RECEIVE_TIME	DATE	Yes

Worklist Audit

Column Name	Data Type	Nullable
CWDOCID	VARCHAR2(16 BYTE)	No
PRIORITY	NUMBER(2,0)	No
OPRIORITY	NUMBER(2,0)	No
DUE_DATE	DATE	No
ODUE_DATE	DATE	No
USER_ID	VARCHAR2(64 BYTE)	Yes
OUSER_ID	VARCHAR2(64 BYTE)	Yes
INITIATOR	VARCHAR2(64 BYTE)	Yes
AUDITDATE	DATE	No
OPERATION	NUMBER(2,0)	No
TAKENWID	VARCHAR2(16 BYTE)	Yes
DESCRIPTION	VARCHAR2(128 BYTE)	Yes

User Defined Statistics

Column Name	Data Type	Nullable
CREATIONDATE	DATE	No
VMID	NUMBER(8,0)	No
HOST	VARCHAR2(64 BYTE)	No
STYPE	NUMBER(2,0)	No
PARAM1	NUMBER(16,0)	No
PARAM2	NUMBER(16,0)	No
PARAM3	NUMBER(16,0)	No
PARAM4	NUMBER(16,0)	No

User Counts

Column Name	Data Type	Nullable
SERVERID	NUMBER(9,0)	No
STARTED	DATE	No
LASTUPDATED	DATE	No
HOST	VARCHAR2(64 BYTE)	No
IPADDRESS	VARCHAR2(22 BYTE)	No
MAXLOGINUSER	NUMBER(6,0)	No
MAXGUESTUSER	NUMBER(6,0)	No

Change History

Column Name	Data Type	Nullable
ID	VARCHAR2(16 BYTE)	No
CWCREATEDBY	VARCHAR2(64 BYTE)	Yes
CWDOCSTAMP	VARCHAR2(9 BYTE)	Yes
LASTUPDATEDDATE	DATE	Yes
PARENTOBJECTID	VARCHAR2(32 BYTE)	Yes
OBJECTLABEL	VARCHAR2(10 BYTE)	Yes
OBJECTTYPE	VARCHAR2(3 BYTE)	Yes
OPERATION	VARCHAR2(1 BYTE)	Yes

Event Log

Column Name	Data Type	Nullable
EVENT_ID	NUMBER(16,0)	No
TRANSACTION_ID	NUMBER(15,0)	Yes
AVM_ID	NUMBER(9,0)	Yes
EVENT_SOURCE	NUMBER(2,0)	No
EVENT_SEVERITY	NUMBER(1,0)	No
EVENT_CODE	NUMBER(4,0)	No
DESCRIPTION	VARCHAR2(1024 BYTE)	Yes
EVENT_TIME	DATE	No
USER_ID	VARCHAR2(64 BYTE)	Yes
METADATA_TYPE	VARCHAR2(256 BYTE)	Yes
METADATA_VER	NUMBER(9,0)	No
OBJECT_ID	VARCHAR2(256 BYTE)	Yes
EXTERNAL_CODE	VARCHAR2(8 BYTE)	Yes
EXTERNAL_TYPE	VARCHAR2(64 BYTE)	Yes
QUALIFIED_NAME	VARCHAR2(256 BYTE)	Yes
STACK_TRACE	VARCHAR2(4000 BYTE)	Yes



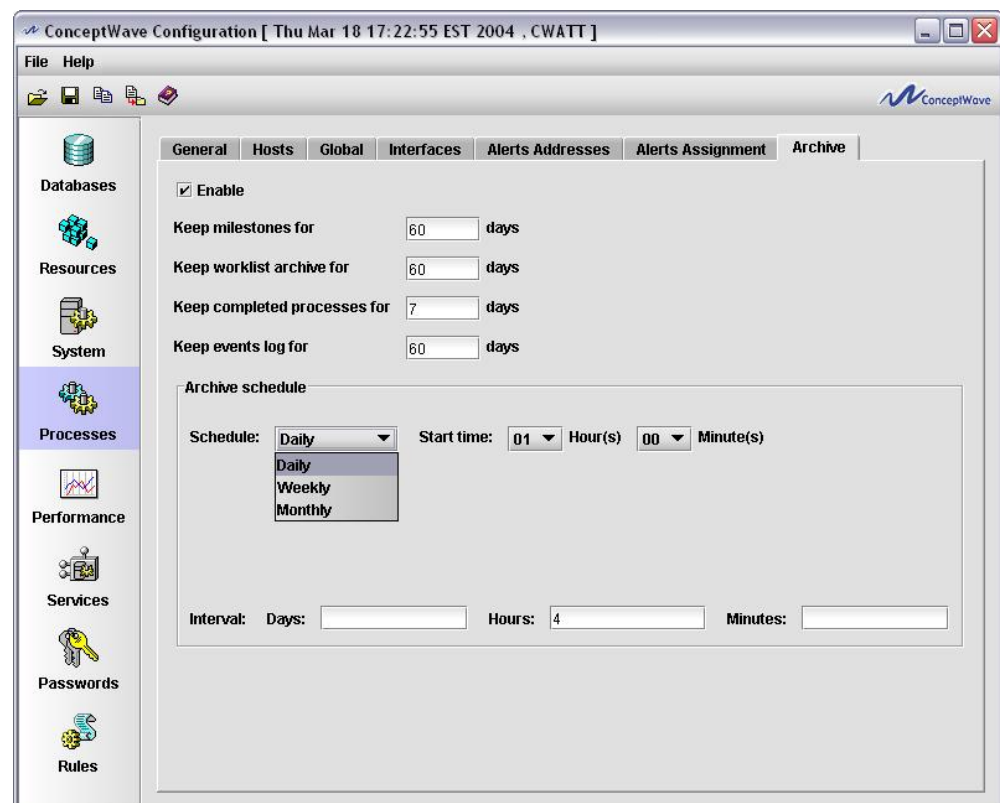
3.8 Archiving

Ericsson Catalog Manager provides "current" and "historical" tables, with auto-archiving to history upon quote completion. Data archived to historical tables is immediately retrievable online. These capabilities cover process related tables that affect online performance:

- Worklist
- Process
- Process Activity

All other data within the system is kept indefinitely, unless Archiving is enabled within the system. The Archiving tab in the Ericsson Catalog Manager Configuration tool provides the means to specify parameters for the removal of certain workflow and system items from the database. The Archiving GUI allows for certain types of items (DB records) to be deleted after a definable period of time using automated scheduling functions. The items that can be managed are:

- Milestones
- Worklist Archive
- Completed Processes
- Events Log





In addition, a system defined rule is available for specifically managing the archiving of orders and its associated data. By invoking this Archive Order Rule and creating an expression with conditions related to its type, age or any other factor, the resultant set of orders returned based on that rule can be forwarded to a data warehouse or data-mart environment and then subsequently deleted from the Ericsson Catalog Manager database.

All business data is stored within Oracle and as such the backup and archiving policies are established via Oracle. Oracle table partitioning, for example by month, is fully supported.

4 Single Sign-On, Authentication and Password Management

4.1 Authentication and Password Management

The Ericsson Catalog Manager solution performs authentication and password management functions through a security provider. Single sign-on can be achieved by having the Ericsson Catalog Manager application securely leverage the existing authentication mechanisms that are in place at the CSP.

4.1.1 Built-in Security Providers

Currently, the following security providers are supported. Additional Security Providers can be developed by Ericsson Catalog Manager on client request and configured by the User Profile Administrator.

4.1.1.1 OpenID

OpenID is a free and open decentralized standard designed to enable user-centric digital authentication. OpenID is in the form of a unique URL, and is authenticated by the identity management server (such as Atlassian Crowd) used by your organization. The OpenID protocol allows you to prove that you are the owner of the URL.

By running an identity server on your site, administrators can configure settings to authenticate user's credentials. This approach consolidates a single user's login credentials, for multiple applications, to a single username and password.

A sample identity management server to use with OpenID is Atlassian Crowd. This server supports the following directories:

- Microsoft Active Directory
- OpenLDAP
- Sun
- Novell eDirectory



- Apache Directory Server

Once you have logged in to the OpenID server, your login can be shared with several OpenID enabled applications in the same Web browser. OpenID is a flexible solution and a de facto standard for integrating disparate applications into portals in the cloud. Applications can be built using different technologies as OpenID support is not limited to Java™ application servers.

4.1.1.2 Kerberos

When using Kerberos SSO, once you have logged in to your LAN or Operating System account, your login and authentication is passed to the Ericsson Catalog Manager application seamlessly. This single sign-on solution is more transparent, but is less flexible as its setup is application server-dependent. Configuration is required on the windows client (browser), and on the application server, e.g. WebLogic or WebSphere.

4.1.1.3 Ericsson Catalog Manager Security Provider

The Ericsson Catalog Manager security provider (effectively Oracle) has the following characteristics:

- The default security provider.
- Uses the Oracle database user accounts for authentication (performs database connection).
- When creating a new user in the User Profile Administrator Tool, a new Oracle database user account is created if one does not already exist. The Oracle database user account has the same password as the user ID.
- When deleting a user from the User Profile Administrator Tool, the corresponding Oracle database user account is deleted. If deleting an Oracle database user account automatically is not desirable, then the User Profile Manager database account (refer to the Ericsson Catalog Manager Installation User Guide, Oracle DB Management) should not be assigned the DROP USER system privilege which allows deletion of Oracle users.
- User Profile Administrator can reset user passwords.
- Users can change passwords using the Ericsson Catalog Manager framework.

4.1.1.4 Windows NT Security Provider

Windows NT security provider has the following characteristics:

- Uses Windows NT domain security for authentication.
- The Windows NT domain used by the security provider is set in the security provider Data field. This domain will be used as the default domain for user authentication.
- Can import Windows NT domain groups and users.
- Users can change passwords using the Ericsson Catalog Manager framework.
- Can only be used on Windows machines.



- Supports multiple Windows domains.

4.1.1.5 JAAS (Java Authentication and Authorization Provider) Security Provider

JAAS security provider has the following characteristics:

- Uses JAAS provider for authentication.
- JAAS login module name is set in the security provider Data field.

4.1.1.6 LDAP (Lightweight Directory Access Protocol Provider) Security Provider

LDAP security provider has the following characteristics:

- Uses LDAP server for authentication.
- The LDAP URL and principal used by the security provider are set in the security provider Data field.

4.1.1.7 SMB Provider

SMB security provider has the following characteristics:

- Uses Windows NT domain security for authentication.
- The Windows NT domain and domain controller used by the security provider is set in the security provider Data field.
- The specified domain will be used as the default domain for user authentication and the specified domain controller will be used as the authentication server. The domain controller is optional, if one is not specified the domain controller is automatically detected based on the specified user domain.
- Can be used on non-Windows machines.
- Supports multiple Windows domains

When the SMB Security Provider is selected, the Network Info menu becomes available from the main GUI. Here a user may access the Domain Controllers submenu to configure domain preferences.

Use the Add or Search buttons to add domains to the list, and set their status by checking Active. Note that after a domain has been added, only its Active status can be changed. To rename a domain, it must be deleted and added again.

4.1.1.8 Other (Additional) Security Providers

Other (additional) security providers have the following characteristics:

- Can be developed and used by the User Profile Administrator Tool by selecting <other> in the security provider combo box.



- Use the Other field to give the Java class of the security provider. For example, com.conceptwave.security. Use the Data field for parameters.
- Multiple Windows Domains Support

Both Windows NT and SMB security providers support user authentication across multiple Windows domains. The Windows NT provider allows specification of the NT domain name when importing users and groups. The user domain is stored as part of the user ID in the User Profile database in the format <domain>\<user>. When creating users manually in the User Profile Administrator Tool (in the case of the SMB provider), the user ID should contain the domain name as specified above. The user domain name is also stored as a separate attribute in the User Profile document defined in the Ericsson Catalog Manager User Profile namespace and is displayed as part of the user name.

During the logon process, the user can enter his/her user ID in full format (i.e. <domain>\<user>) or the user name only (i.e. <user>). If the user enters the user ID without including the domain, the default domain is used to authenticate the user.

4.2 User Privilege Management

In order to allow certain functional tasks to be performed only by specific users with certain skills, Ericsson Catalog Manager allows the assigning of user privileges to user participants. User privileges specify allowable actions in the Ericsson Catalog Manager system. User privileges are defined and assigned to the user in the User Profile Manager.

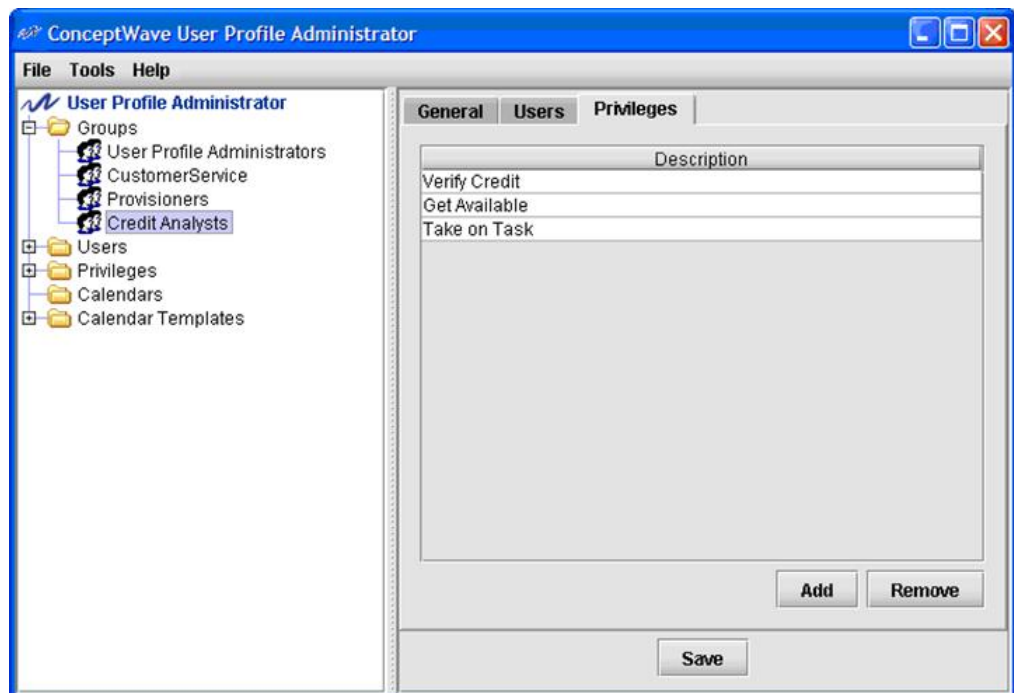
Privileges are used in the following manners:

- To define a privilege that must be held in order to access an Ericsson Catalog Manager application
- To define a privilege that must be held in order to view or perform a specific action
- To identify the orders, order pages, forms and fields that are visible and/or editable
- To identify membership within a “participant” or workflow actor
- To define a privilege that must be held for external systems to communicate to Ericsson Catalog Manager via an API



The relationship between Users and the privileges they hold are defined using “Groups” so that:

- A group holds privileges
- A user may be a member of one or more groups
- The user holds the sum of all privileges held by all the groups to which he/she belongs



Groups do not necessarily map directly to the business functions or working groups within an organization. A single operational group may be represented by more than one User Profile Group. Examples include, but are not limited to:

Operational groups where the set of functions performed are further broken down. For example, an “Installer” and an “Expert Installer” group may be defined for the same operational group. The “Expert Installer” group may have additional privileges that allow them to deal with a special type of install, or permit them additional update privileges within the system.

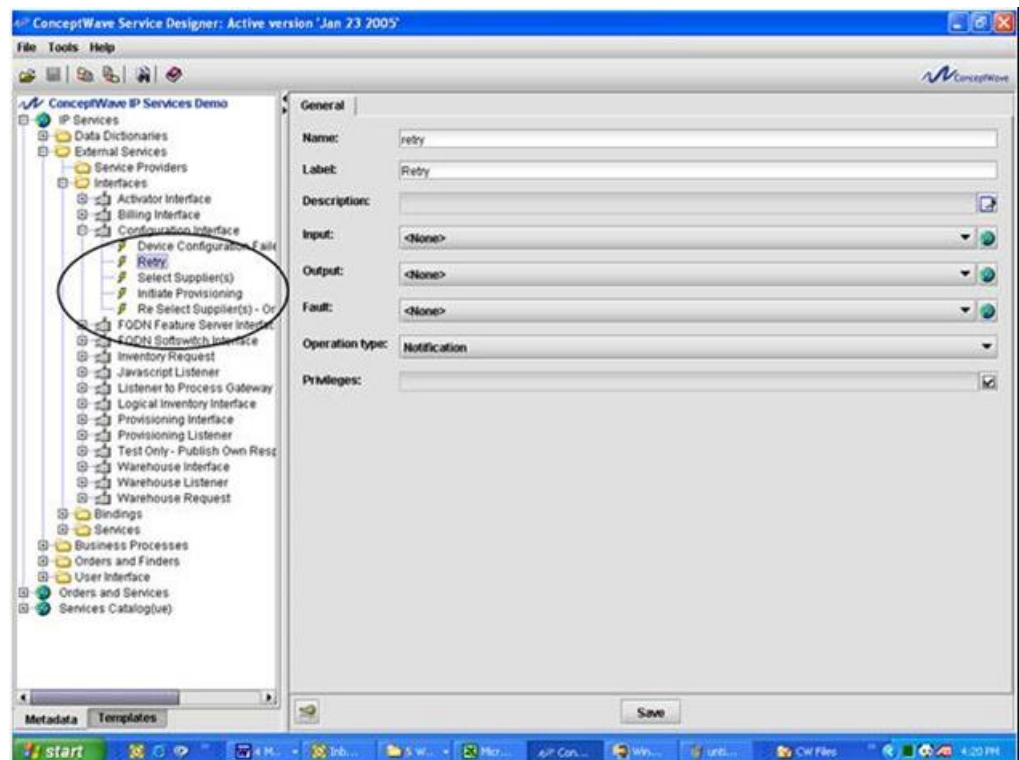
Operational groups where the group operates on different Calendars, i.e. across multiple time zones, different shifts, etc. In this case, the Groups hold identical privileges but are associated with different Calendars to represent their differing availability

For this reason, there is not a direct one-to-one relationship between the User Profile Group, and the “Participant” (actor, workflow workgroup, etc.).

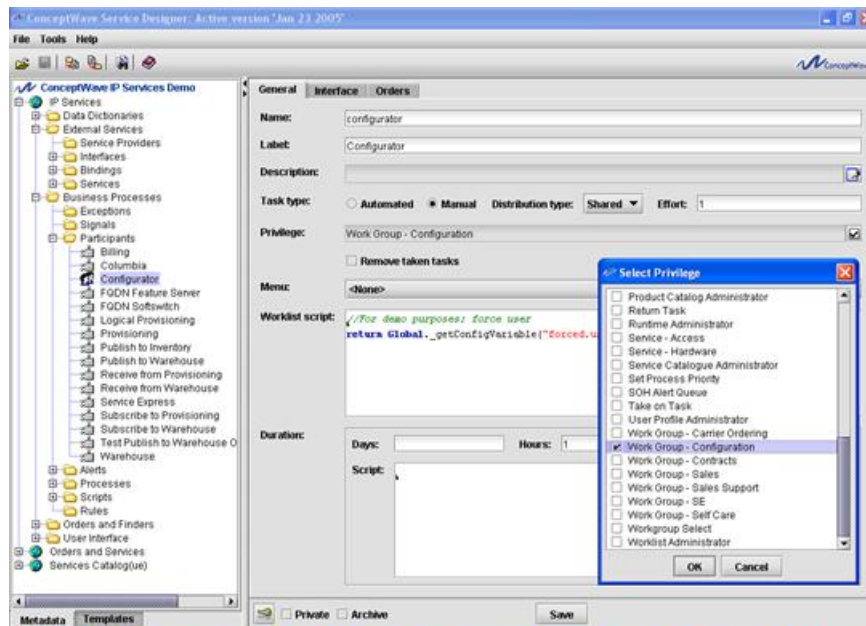
From a metadata/process definition standpoint, there are a few items required to be configured in the metadata to support this high level of flexibility and granularity.



- **Interfaces:** Manual/Human interfaces are defined to represent the set of interactions between the workflow and the “Participant”. Each interface is composed of a set of operations...
- **Operations:** The system needs to understand what types of operations can be assigned to or performed by each interface. For the purposes human participants there are two types of operations: one-way and notification. A one-way operation is used to assign a task to a participant. The system sends a task to a participant, but does not expect instant feedback. When the human participant performs an operation, the human participant notifies the system that he/she has done so. This type of operation is a “notification”. Each operation may optionally carry a Privilege that must be held in order to perform the action (more on this later).



- **Participants:** ‘Actors’ that interact in the workflow are defined and assigned to an Interface definition (the set of operations to be performed). It is possible to restrict the operations within the Interface that may be performed by the participant using permissions/privileges.
- **Privilege:** A Privilege is assigned to a Participant to represent the privilege that must be held to be able to work as the Participant/Actor (to be assigned work items destined for this Participant).



Some observations:

- Multiple Participants may be assigned to the same privilege or a unique privilege may be created for each Participant. The latter approach is recommended to maximize the flexibility to redistribute work across User Profile groups (by maximizing the granularity of the permissions).
- Certain operations may be made available for a subset of the members of a Participant by defining an additional privilege that must be held to perform the additional operations. In this case, a second User Profile Workgroup would be defined to hold the additional privilege.
- A single User Profile Group may be associated with multiple Participants by granting a set of privileges. This permits the aggregation of workflow actors into a single operational group while retaining the granularity within the workflow. This maximizes the organization's ability to redistribute work without affecting the workflow or workflow actor definitions.
- Since users may have a number of user privileges assigned to them, users can perform any user participant task (operation) according to the privileges that he/she possesses. In addition to specifying functional tasks to users belonging to a Group skill level, the system can be configured to present only the relevant, personalized, dynamic and real-time content and information necessary to perform the tasks allocated to them.

During the metadata configuration process (via the Service Designer/Velocity Studio Tool), a "permissions" tab is available for each order, order page/document, form/view and field to specify the users that have access to that information.



5 Adapters

An adapter is a layer of software that converts data from an application into a common form that can be understood and used by other applications. Adapters make it easier to integrate two different applications, without the need to understand and work with the complexities of the two applications.

5.1 Competitive Advantage

Interface adapters allow communications service providers to successfully compete in chaotic and uncertain economic times by changing and integrating with rapidly evolving technologies.

Two of the most painful, expensive problems facing communications service providers' Operations Support Systems (OSSs) are multivendor interoperability issues and system adaptability. These problems are often obstacles for communications service providers to successfully deploy next generation networks and to rapidly introduce new services.

Interoperability and system adaptability will become even more vital in the coming wave of consolidation among telecommunications and Internet service providers. Carriers looking to offer new services or enter new markets are opting to merge, acquire, or partner with other carriers, rather than grow organically. Such options may speed up market entry, but wreak havoc on both Business and Operations Support Systems, and on the ability to adapt and change in a chaotic world.

5.1.1 Advantages of Ericsson Catalog Manager Adapters

Adapters help communications service providers overcome these challenges and gain a competitive advantage by enabling interoperability between different systems.

With Ericsson Catalog Manager Adapters, you can:

- Have control over your data model and business rules
- Reduce the impact of complex integration and data migration by employing adapters that enable the translation and communication of messages
- Eliminate interoperability and system adaptability issues by using custom-designed adapter solutions

Custom-designed adapters ensure that messages between Ericsson Catalog Manager modules and your external system are translated, and are sent accurately and securely. Ericsson Catalog Manager hides the complexity of having the calling system interpret low-level APIs that force the calling system to understand the other system's business logic and rules, allowing the process modeler to easily model the interface and messages.



5.1.2 Key Benefits:

- Have complete control over your data model and business rules, to reduce complex integration and data migration
- Eliminate interoperability and system adaptability issues by using custom-designed adapter solutions
- Quickly deploy and introduce new products

5.2 Orga OPSC Gold Adapter

5.2.1 Introduction

Ericsson Catalog Manager provides a custom adapter for Orga OPSC Gold, which is a convergent real-time rating, charging and billing system. OPSC Gold has a modular platform architecture, which makes it a very versatile system both for adjunct rating uses cases and for end-to-end convergent billing solutions. It can either enhance existing billing systems or replace them as an independent single system.

The Orga OPSC Gold adapter exposes the functionalities provided by Orga OPSC Gold and allows other systems to effortlessly interact with Orga OPSC Gold.

5.2.2 OPSC Gold System Modules

The main modules of the OPSC Gold System are:

- **Administration and Configuration Cockpit (ACC)** – Eclipse based GUI of OPSC Gold
- **Business Administration System (BAS)** – business logic for customer, account, product and tariff management. BAS includes the Operations Control Manager (OCM) component for execution of configurable business logic
- **Configuration and Operation Module (COM)** – the COM provides the basic configuration data and configuration framework of OPSC Gold
- **Payments File Processor (PFP)** – the PFP provides a batch interface in order to process incoming payment files from the Payments Interface
- **Business Intelligence Support Module (BISM)** – the BI Support Module is the central reporting framework and staging area of OPSC Gold for providing reporting data to external systems
- **Network Interface Module** – provides Unified Provisioning Interface (UPI) for transformation of provisioning requests and Post Event Converter (PEC) for conversion of usage events
- **Real-time Environment (RTE)** – real-time and post-event rating based on InCore in-memory database
- **Core Billing Framework (CBF)** – event import, bill calculation and bill extraction processing, payments and journaling



5.2.3 Business Service Gateway (BSG)

OPSC Gold has a standard interface to external business support systems and to Orga Systems' Business Service Gateway (BSG), which provides a Web Services Application Programming Interface (API) and Batch File Interface for OPSC Gold.

5.2.4 Overview of the BSG Web Services API

The BSG Web Services API can be used by third-parties to integrate with BSG. The overall API is categorized into two broad functional categories: Informational operations and Modification operations.

- The Informational operations are used to query the OPSC Gold to retrieve necessary data records and do not change anything in OPSC Gold.
- The Modification operations are used to change data in OPSC Gold.

5.2.5 Supported BSG Functionalities

The OPSC Gold adapter provides functionalities for storing, using, and manipulating customer, account, and subscriptions data. It also has functionalities for dealing with Product Lifecycle Management (PLD) and Inventory. The following are some of the examples of the supported functionalities:

- Customer
 - Create customer
 - This operation is used to create a new customer. This creates a customer with the customer address.
- Account
 - Create top-level account
 - This operation is used to create top-level account. This means that a new account hierarchy is created for a customer. The account will be created with the life cycle state "deployed".
 - Get account
 - This operation is used to retrieve all necessary information about the accounts and subscriptions from the database and back-ends.
 - Get account balances
 - This operation is used to retrieve information about account balances from the database and back-ends.
 - Update account
 - This operation is used to update an existing account or subscription. The service will update either a top-level or child account or subaccounts.
 - Terminate account
 - This operation is used to terminate an account.
 - Get account bills
 - This operation is used to retrieve information about the bills associated with an account.

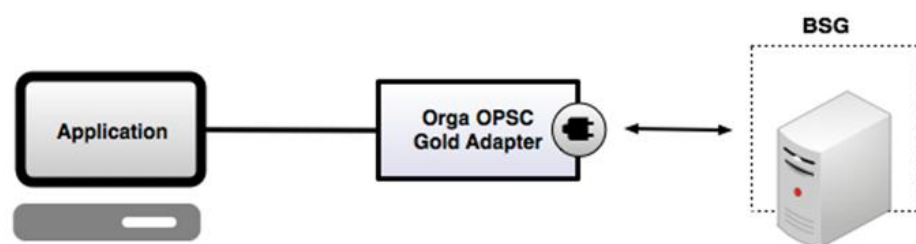


- Subscriptions
 - Create subscription account
 - This operation is used to create a subscription account. The service appends a subaccount as a child to an existing account (top-level or child). The subscription will be created with the life cycle state “deployed”.
- Billing
 - Charge one-time fee
 - This operation is used to charge a one-time fee in an account. It also allows the overwriting of the charged price.
 - Top-up account balances
 - This operation is used to top-up account balances.
 - Top-up is a feature that OPSC Gold provides natively for any balance that is available to an account.
- Products
 - Add sold products
 - This operation is used to add new sold products to the accounts.
 - The included sold components will be created with the life cycle state “deployed.” The life cycle progression will be stored as scheduled changes.
 - Deactivate multiple products
 - This operation is used to deactivate multiple products.
- Events
 - Search event records
 - This operation is used to search for all event records which fulfill the given search criteria and filters, and then return those records.

5.2.6

CGS Billing Adapter Connectivity

Orga OPSC Gold Adapter Connectivity Diagram:





5.3 CSG Billing Adapter

5.3.1 Introduction

Ericsson Catalog Manager provides robust, flexible, and adaptable custom-built adapters for CSG International's Billing system. The CSG Billing system offers fully integrated modules for rating, discounting and bill production to help companies maximize their investment and optimize their business. The CSG Billing adapter exposes the functionalities provided by CSG Billing and allows other systems to effortlessly interact with CSG Billing.

5.3.2 Custom-Built Adapters

Ericsson Catalog Manager is able to provide robust custom-built adapters which allow different applications to interact with and use the CSG Billing system. The advantage of a custom approach is that we can satisfy each of our customer's specific needs and fully comply with their particular implementation of the CSG Billing system. Having custom-built adapters ensures that the adapter only supports the functionalities that are actually being used by a particular customer; rather than being bloated with functionalities which may not be needed by the customer.

5.3.3 Working with World-class Customers

Ericsson Catalog Manager has already provided its reliable custom-built CSG Billing adapters to some of the biggest telecommunications companies from around the world, including Time Warner Cable, Comcast, and Charter.

5.3.4 Supported CSG APIs

Ericsson Catalog Manager's custom adapters are able to support the CSG APIs needed by our customer, such as CSG's SODI (Service Order Delivery Interface) API set.

5.3.5 Supported Functionalities

The Ericsson Catalog Manager CSG Billing adapter is able to support any functionalities required by our customer. The specific functionalities supported in a particular version of the adapter depend on the customer's implementation of the CSG Billing system. Examples of the functionalities which can be supported by the CSG Billing adapter include the following:

- Customer
 - Create new customer
- Account
 - Add account
 - Retrieve account



- Order
 - Create order
 - Update order
 - Verify order
 - Fallout Handling
- Location
 - Find location
- Jobs
 - Get Jobs Available Schedule

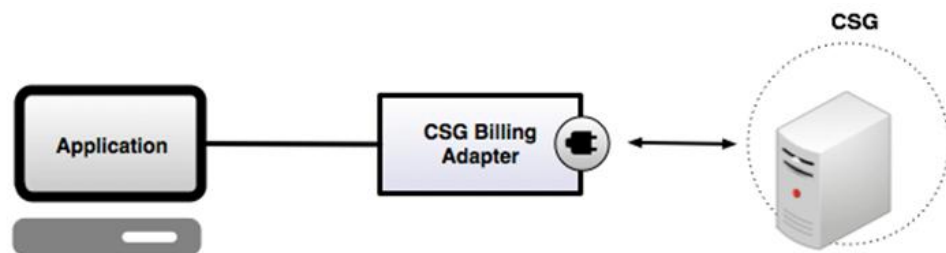
5.3.6 CSG Adapter Data Organization

Ericsson Catalog Manager can define different data models to fit our customers' needs. An example of a data model that can be implemented by the CSG Adapter is shown below.

- Keys
- Response
- Customer
- Account
- Location
- Order
- Items
- Equipment
- Jobs

5.3.7 CGS Billing Adapter Connectivity

CSG Billing Adapter Connectivity Diagram:





6 Standards Adoption and Adherence

Ericsson Catalog Manager's framework is a J2EE compliant n-tier JAVA application which adopts open standards. WSDL is used for interface definition, XML for metadata repository, BPML for workflow, XSD for data definitions, CSS for Web presentation style, JDBC for data access and JavaScript for customizations and business rules. A zero footprint browser enabled client, non-reliance on any specific J2EE application server or database vendor, and a JRE enabled design time component further differentiates Ericsson Catalog Manager as the ideal service delivery automation vehicle.

6.1 Solution Architecture

Ericsson Catalog Manager provides a solution that splits the presentation, processing and database tier, as well as exposing design, configuration, and administration layers.

6.1.1 Presentation Tier

Ericsson Catalog Manager provides a browser-based presentation layer. As such, it is a true "zero footprint" client with no software, no applets or plug-ins required for clients. This common interface can be used by CSP's customers, channels and internal users responsible for creating, updating, viewing or reporting in the application. This implementation of a simple and intuitive common Web interface for all users simplifies the training requirements and eliminates the need for special training. A key strength of the solution is the auto-generation of the UI (i.e. without resorting to hand painting forms in a Web publishing tool) based upon the underlying metadata.

6.1.2 Processing Tier

The Framework provides the user interface, business rules, and integration infrastructure for the application. This component is essentially the application accessed by the end-users via their browser.

The Process Engine executes the workflow in the background. Provides no user interface itself though the status of the process and the ability to suspend, resume and terminate processes is provided through the Administration Application (see below).

6.1.3 Database Tier

All business and configuration data is stored within the Oracle database.



6.1.4 Design

Service Designer (SD) and Velocity Studio (VS) are the primary configuration tools used to create the application. The data models, business rules, workflow, user interface, interfaces, etc. are all defined within this tool. It generates the metadata (an XML document containing the configuration information) that is used by the run-time applications.

Service Designer/Velocity Studio provides a translation facility within which all menus, forms, error messages and pick-lists are translated into the target languages. It generates language specific resource files (XML documents) that are used by the run-time applications.

6.1.5 Configuration

System Configuration provides a mechanism for specifying the environment specific (Development, Integration Testing, UAT, Production etc.) attributes under which the application is to operate. Examples include logging levels, database location, schemas and passwords, server addresses of systems being interfaced with, cache sizes, polling intervals, etc.

The User Profile Administration Tool allows for the creation of users, groups, privileges and for the management of calendars (holidays, working hours, etc.).

6.1.6 Administration

Run time administration tools for the framework and the process engine instances (AVMs) consist of a native browser based UIs, and a JMX based monitoring/change facility that can be accessed via any JMX enabled management platform.

A multitude of views and reports are provided including: list of AVMs running, the memory utilization, users logged in, message queue sizes, licensing information, process instance status, work queues, event logs, etc. A process administrator has the ability to suspend, resume and terminate processes from this application.

6.2 J2EE Architecture

The Ericsson Catalog Manager product is a J2EE compliant n-tier architecture Product. Java throughout.

- Browser enabled client. Zero footprint.
- No reliance on specific J2EE app. server
- No reliance on specific database vendor
- JRE enabled design time component



The whole Ericsson Catalog Manager suite is developed in Java 2 and uses the services delivered by the Java 2 Enterprise Edition (J2EE). Hence, it is Operating System independent and can run on Windows, Solaris, HP-UX, AIX and Linux.

Ericsson Catalog Manager can utilize the leading Web servers such as Apache and Microsoft IIS.

The internal database server is Oracle but the platform can communicate to any database with a JDBC driver.

End-users access the application using a browser, either MS Internet Explorer or Firefox.

6.3 Service Oriented Architecture

Ericsson Catalog Manager's metadata driven approach enables clients to deploy applications that are Service Oriented Architecture (SOA) compliant¹:

- Standardized Service Contracts
- Service Loose Coupling
- Service Abstraction
- Service Reusability
- Service Autonomy
- Service Statelessness
- Service Discoverability
- Service Composability
- Service-Oriented and Interoperability

Ericsson Catalog Manager meets these SOA Principles through:

- Metadata driven interface definition that allows for any internal function to be exposed out through an interface.
- The provision of standard APIs (service contracts) that may be further tailored through metadata to meet the client's needs.
- The provision of metadata driven interface modeling that allows the client to define its own reusable, autonomous and stateless services.
- Publishing WSDLs for others to auto-discover
- The adoption of Web Services Definition Language (WSDL) in interface modeling

¹ Source: www.soapprinciples.com



6.4 Web Services and Integration

Ericsson Catalog Manager employs WSDL, an XML based standard that defines an interface in 7 layers, to model APIs.

- Systems or “services” can be defined to be supported on a variety of “ports”. Each port specifies a physical location for the service, and a “calendar” which describes hours of availability. This allows for the definition of alternate system access in the event the primary targeted system is unavailable, busy or has failed.
- A system’s interfaces, i.e. the operations it supports, and the data exchanged in each operation are defined. Support is provided for specifying the outbound, inbound (successful) and inbound (error) messages.
- Each port specifies a “binding”, which binds the port to the interface and the Service Provider (technology or protocol) to be employed.

APIs are modeled using WSDL whether they are Web services offered to others, or the definition of Web services that are to be consumed by the system.

Ericsson Catalog Manager supports the importing of WSDL document definitions to the metadata through the Service Designer tool to create data structures (messages), interfaces with operations, bindings and external services with ports based on WSDL document definitions.

Ericsson Catalog Manager supports the importing of XSD schema, and the creation of XSD schema from XML data, through the Service Designer tool to aid in the definition of interface messages.

SOAP messages are processed and depending on the specified business rules, generate internal events that are handled by the Ericsson Catalog Manager solution or external events that are redirected to external bus.

6.5 BPM/Workflow

The Ericsson Catalog Manager workflow engine is a BPML (Business Process Modeling Language) based workflow engine. Ericsson Catalog Manager was one of the initial members of BPMI.org before it was absorbed by the Object Management Group. Our adoption of their BPML workflow specification demonstrates Ericsson Catalog Manager’s commitment to standards adoption.

The Workflow engine contains a rules language, JavaScript, that permits the process modeler to leverage Ericsson Catalog Manager’s rules engine, data transformation facilities and order validation services in order to populate interface messages, process response data, define and evaluate the path to follow in scripts, define the compensate activities, activity durations, etc.

Extensions are provided via JavaScript exits that are defined for each activity.



6.6 TeleManagement Forum: NGOSS, eTOM, TAM and SID

Ericsson is a member of the TeleManagement Forum and a strong supporter of Frameworkx, NGOSS and its three cornerstones: the enhanced Telecom Operations Map (eTOM), SID, and Technology-Neutral Architecture.

Ericsson Catalog Manager's modules are SID-certified. For details on the SID certification and levels attained, see the following Frameworkx Compliance section.

Ericsson Catalog Manager inherently has all the building blocks to integrate within Technology Neutral Architecture (TNA). Ericsson Catalog Manager's approach to metadata definition of interfaces, their operations and terminations, etc. facilitates a Communications Service Provider's adoption of a TNA strategy.

6.7 Frameworkx Compliance

6.7.1 What is Frameworkx

TM Forum's Frameworkx Integrated Business Architecture is a suite of industry standards that provides the blueprint for Service providers (SPs) to enable effective business operations and assess and improve performance. Frameworkx uses a proven, service-oriented approach to rationalize IT operations and integration, which allows SPs to focus on growing their businesses through cost reduction and business agility improvement.

Frameworkx is a result of ongoing collaboration development by TM Forum's member companies from across the industry. Driven by SP requirements, Frameworkx is constantly evolving along with market needs and changes. Frameworkx has been adopted by 90% of the world's largest SPs.

There are four key components within Frameworkx which enables a service-oriented, highly automated and efficient approach to running a SP business:

- 1 Business Process Framework (eTOM) – the industry standard process architecture for both business and functional processes
- 2 Information Framework (SID) – a common reference model for enterprise information that service providers, software providers, and integrators use to describe management information
- 3 Application Framework (TAM) – which provides a common language between SPs and their suppliers to describe systems and their functions, as well as a common way of grouping them
- 4 Integration Framework – which delivers a service oriented integration approach with standardized interfaces and support tools



The key business benefits in using Frameworkx are:

- Speed deployment of new services with industry-proven technologies, business services and platforms
- Reduce integration time and product costs through standard processes and interfaces
- Support new technologies and business models readily
- Reduce risk drastically with well-defined interfaces and certified compliant products
- Leverage industry concepts including SOA and ITIL to provide future-proof solutions

For more information on TM Forum and Frameworkx, please visit www.tmforum.org.

*Reference: TM Forum

6.7.2 Compliance and Conformance

TM Forum provides a Frameworkx Conformance Certification service whereby service providers and suppliers can submit self-assessments of solutions implemented by the service providers and products developed by vendors to TM Forum for evaluation by TM Forum. Upon completion of the assessment requirements, the products (or solutions) are certified by TM Forum as being Conformant. The results are published by TM Forum on their Web site. The Frameworkx conformance certification can be under the Business Process Framework (eTOM), Information Framework (SID) or both.

Ericsson, prides itself as a world-class product vendor that meets the Frameworkx standards and strides to evolve its products to meet the ever-changing standards and market demands. Ericsson Catalog Manager ensures its product and modules are compliant to standards from initial design through to development, which is a primary business objective.

Ericsson has been engaged with TM Forum for many years. It was a pioneer with the formal conformance certification program with TM Forum in 2009. Ericsson Catalog Manager continually conducts informal self-assessment against the standards during our product management and design phase, and is fully confident that it can demonstrate the products are in compliance with Frameworkx standards.

While being compliant to standards, it is important to remember that market leadership is not gained with standards alone. Ericsson Catalog Manager has been able to excel and deliver that world-renowned product by ensuring that solutions meet customer requirements and excel beyond industry standards.



6.7.3 Current Assessment

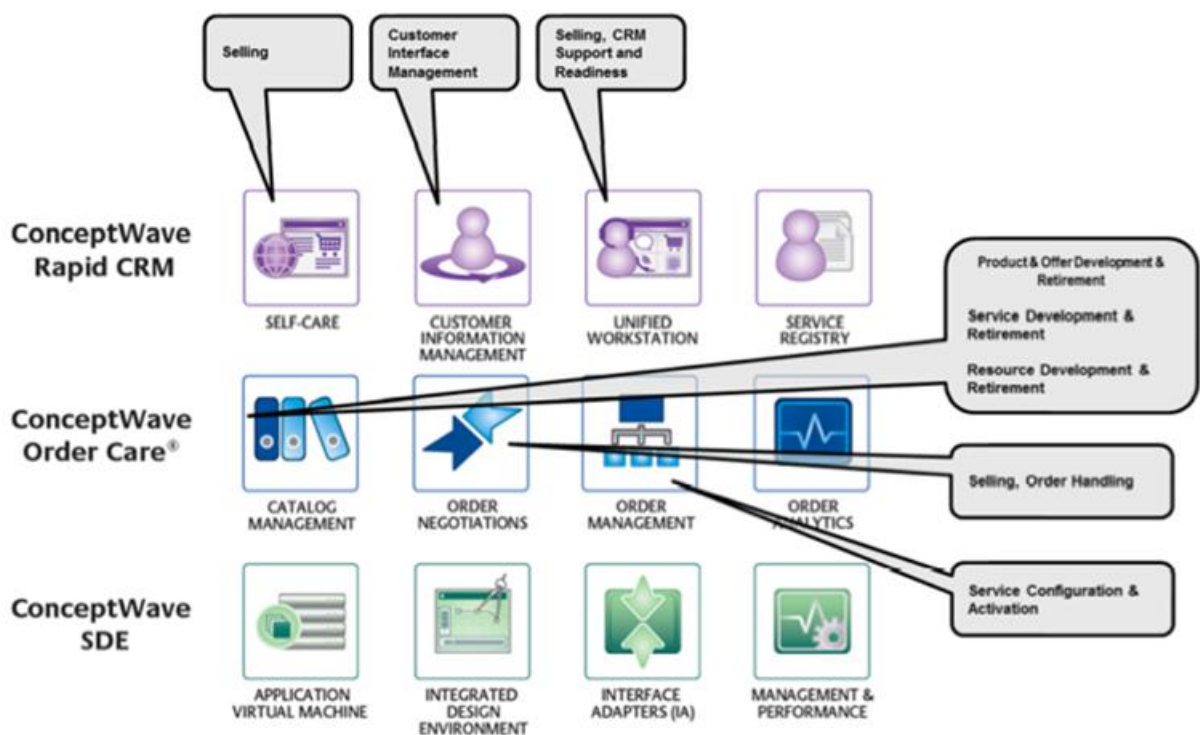
Current Ericsson Catalog Manager Assessments can be downloaded from the TM Forum Web site:

<http://www.tmforum.org/ProductAssessmentResults/ConceptWaveOrderCare/12745/home.html>

<http://www.tmforum.org/ProductAssessmentResults/ConceptWaveCatalog/12760/home.html>

Please contact marketing@conceptwave.com for more information.

6.7.4 Ericsson Catalog Manager's Framework Compliance



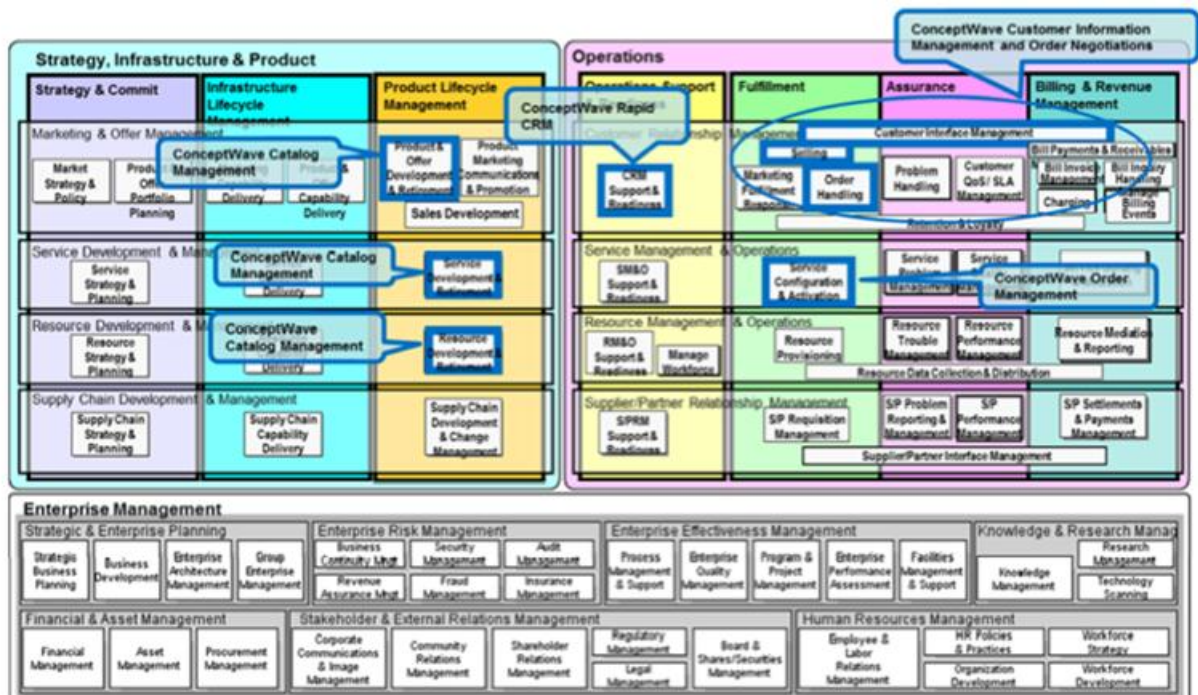


Figure 1 Release 8 of the Business Process Framework (eTOM)

Ericsson Catalog Manager products have been designed to take full advantage of TM Forum's Framework standards. Under Business Process Framework (eTOM), CW products support the following Level 2 eTOM processes:

- 1 1.1.1.1-CRM Support & Readiness
- 2 1.1.1.2-Customer Interface Management
- 3 1.1.1.4-Selling
- 4 1.1.1.5-Order Handling
- 5 1.1.2.1-SM&O Support & Readiness
- 6 1.1.2.2-Service Configuration & Activation
- 7 1.1.3.1-RM&O Support & Readiness
- 8 1.1.3.2-Resource Provisioning
- 9 1.2.1.5-Product & Offer Development & Retirement
- 10 1.2.1.6-Sales Development
- 11 1.2.2.3-Service Development & Retirement
- 12 1.2.3.3-Resource Development & Retirement



L1 Information Framework (SID)

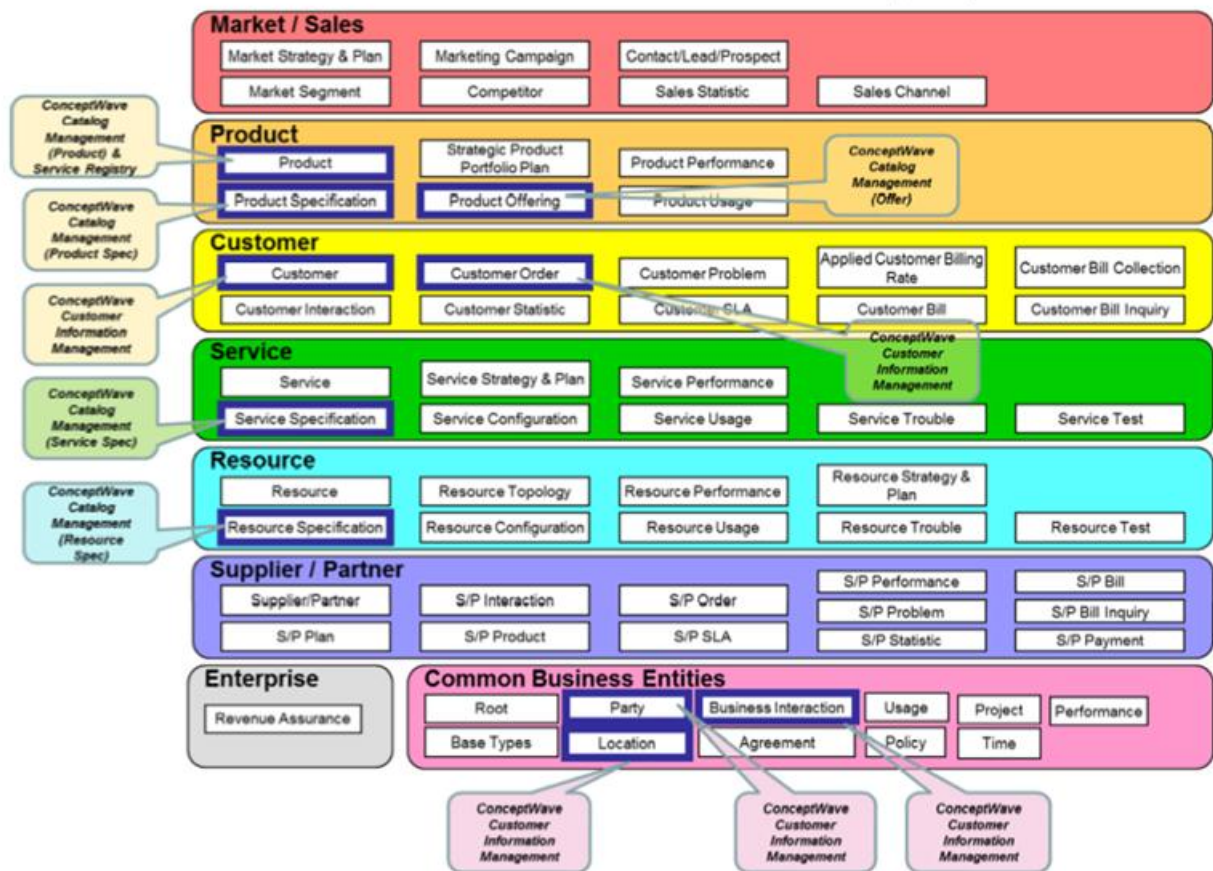


Figure 2 Release 9 of the Information Framework (SID)

6.7.5 Ericsson Catalog Manager's Information Framework (SID) Conformance Certification 2010

Ericsson Catalog Manager has successfully completed the TM Forum's Information Framework (SID) Conformance Certification assessment for its Catalog Management product in 2010. This demonstrates Ericsson Catalog Manager's commitment to provide the best-in-class product that is also conformance to the widest recognized industry standards. Ericsson Catalog Manager Catalog Management-5 is certified by TM Forum to meet Level 7 conformance for the following SID ABEs:

- 1 Product / Product / Product Order
- 2 Product / Product Price / Pricing
- 3 Product / Product Offering / Offerings
- 4 Product / Product Catalog / Catalog and Domains
- 5 Product / Product Offering Price / Pricing
- 6 Product / Product Specification / Products and Components

The detail of the SID conformance certification result is available at the TM Forum:

(<http://www.tmforum.org/CatalogManagementVersion/8814/home.html>)



6.7.6 Ericsson Catalog Manager's Framework Conformance Certification 2011

Ericsson Catalog Manager has completed assessment of its products through the TM Forum Framework Product Conformance Certification program in 2011.

The scope of the assessment includes both the Business Process Framework (eTOM) and Information Framework (SID) components of the Framework standards. The overall program is divided in multiple phases and completion is expected before the end of 2011.

Table 1 Business Process Framework (eTOM) Scope:

eTOM Level 3 Processes	eTOM Level 4 Processes
1.1.1.1 - CRM - Support & Readiness	1.1.1.1.1 - Support Customer Interface Management
	1.1.1.1.2 - Support Order Handling
	1.1.1.1.7 - Support Selling
	1.1.1.1.10 - Manage Customer Inventory
	1.1.1.1.11 - Manage Product Offering Inventory
	1.1.1.1.15 - Support Bill Inquiry Handling
1.1.1.2 - Customer Interface Management	1.1.1.2.1 - Manage Contact
	1.1.1.2.2 - Manage Request (Including Self Service)
1.1.1.4 - Selling	1.1.1.4.1 - Manage Prospect
	1.1.1.4.2 - Qualify Opportunity
	1.1.1.4.3 - Negotiate Sales/Contract
	1.1.1.4.4 - Acquire Customer Data
	1.1.1.4.5 - Cross/Up Selling
1.1.1.5 - Order Handling	1.1.1.5.1 - Determine Customer Order Feasibility
	1.1.1.5.2 - Authorize Credit
	1.1.1.5.4 - Track & Manage Customer Order Handling
	1.1.1.5.5 - Complete Customer Order
	1.1.1.5.6 - Issue Customer Orders
	1.1.1.5.7 - Report Customer Order Handling
	1.1.1.5.8 - Close Customer Order
1.1.2.1 - SM&O Support & Readiness	1.1.2.1.1 - Manage Service Inventory
	1.1.2.1.2 - Enable Service Configuration & Activation



eTOM Level 3 Processes	eTOM Level 4 Processes
1.1.2.2 - Service Configuration & Activation	1.1.2.2.3 - Track & Manage Service Provisioning
	1.1.2.2.4 - Implement, Configure & Activate Service
	1.1.2.2.7 - Issue Service Orders
	1.1.2.2.8 - Report Service Provisioning
	1.1.2.2.9 - Close Service Order
1.2.1.5 - Product & Offer Development & Retirement	1.2.1.5.5 - Develop Detailed Product Specifications
	1.2.1.5.6 - Manage Product Development
	1.2.1.5.7 - Launch New Products
1.2.2.3 - Service Development & Retirement	1.2.2.3.4 - Develop Detailed Service Specifications
1.2.3.3 - Resource Development & Retirement	1.2.3.3.4 - Develop Detailed Resource Specifications

Table 2 Information Framework (SID) Scope:

SID Domain	SID ABE
Common Business Entity	Location/Geographic Place/Geographic Place
	Party/Party
	Party/Contact
	Party/Identification
	Party/Roles
Customer	Customer
*Product	*Product/Product
	*Product/Product Price
	*Product Offering
	*Product Offering/Product Catalog
	*Product Offering/Product Offering Price
	*Product Specification
Common Business Entity	Business Interaction
	Root Business Entities
Customer	Customer Order
Service	Service Specification
	Service Specification/CustomerFacingServiceSpec



SID Domain	SID ABE
	Service Specification/CustomerFacingServiceSpec/CustomerFacingServiceSpec Role
	Service Specification/CustomerFacingServiceSpec/Service Package
	Service Specification/ResourceFacingServiceSpec
	Service Specification/ResourceFacingServiceSpec/ResourceFacingServiceSpec Role
	Service Specification/ResourceFacingServiceSpec/Service Bundle
Resource	Resource Specification
	Resource Specification/CompoundResource Specification
	Resource Specification/LogicalResource Specification
	Resource Specification/LogicalResource Specification/Logical Role Specification
	Resource Specification/Physical Resource Specification
	Resource Specification/PhysicalResource Specification/Physical Role Specification

List of processes and ABEs subjected to changes without notice.

6.7.7 CASE STUDY: TM Forum Frameworkx Used to Re-engineer and Transform Türk Telekom's Business

Türk Telekom's solution was submitted and was selected as the TM Forum Solution Excellence Award Finalist 2011. Detail of the submission is available at the TM Forum:

(<http://www.tmforum.org/ConceptWaveOrderCare/10627/home.html>)

Türk Telekom undertook a major business transformation program called Program Bir to create a customer-centric business, which consists of Customer Relationship Management (CRM) and Service and Resource Management (SRM). They used a new architectural approach and leveraged TM Forum's Frameworkx. Türk Telekom used Ericsson Catalog Manager products to transform their quote-to-cash business process fulfillment and order management requirements.

Ericsson Catalog Manager's SID-based Catalog Management and eTOM-based Order Management solution enabled Türk Telekom to:

- Rapidly deploy Ericsson Catalog Manager's solution that unified all services and subsidiaries
- Automate customer and order lifecycle management
- Improve customer experience through fast and accurate order negotiation and completion
- Deliver quick time-to-market for current and new products and services
- Reduce overall operational costs
- Speed integration between new and legacy systems



7 Product Management

7.1 Software Release Strategy

Ericsson Catalog Manager regularly enhances its products to ensure continuous quality improvements and inject new features and technologies into the core platform. Ericsson is available to help you plan and rollout upgrades quickly, effectively and seamlessly.

There are several types of releases:

- **Major Releases:** Ericsson updates and enhances its Ericsson Catalog Manager product portfolio on a continual basis and delivers these enhancements via scheduled GA (major) product releases twice within a rolling 12 month cycle. Selected major releases warrant longer lead times. Major releases are labeled #.#.0.0, e.g. 4.2.0.0
- **Service Packs:** Service Packs typically contain new features deemed useful in support of active customers and projects. These changes are limited in scope but are of sufficient business benefit to our clients that they warrant releases in advance of the next major release. Service Packs also contain minor product fixes that had workarounds and therefore were not included in an EBF (below). Service Packs are targeted for every 6-8 weeks. Service Packs are labeled #.#.#.0, e.g. 5.1.9.0
- **Emergency Bug Fixes (EBF):** EBFs contain product fixes to issues of sufficient importance that they warrant immediate attention. EBFs are labeled #.#.#.#, e.g. 5.1.0.6.

Software upgrades are provided as part of Software Support Services. Ericsson will work with you to determine how to proceed with a software upgrade. Releases need to be implemented within a reasonable time frame after their release, in accordance with the standard Software Support Services terms.

7.2 End-Of-Life Policy

Products reach their end-of-life for many reasons. Ericsson understands that planning for a product end-of-life milestone is an important part of any organization. Ericsson provides this policy, to help customers plan for end-of-life and end-of-sale milestones, and migration options to other Ericsson Catalog Manager product versions or technology.

The general guidelines are:

- 1 An end-of-life decision will be made after two subsequent major releases have been issued.
- 2 A 6 month notice will be provided for end-of-sale and end-of-life products. This notice will be sent to customers, partners, and will be posted on the Ericsson Catalog Manager support portal.



- 3 Access to Ericsson Catalog Manager Support will be available as per the contracted support terms until the end-of-life milestone for the product in question.
- 4 Customers need to ensure that they have a current and fully paid support contract with Ericsson. Please contact your Account Representative or Ericsson Support for details about your support status.

Ericsson frequently supports clients on releases that have been formally declared to be End-of-Life. This support is provided on a case-by-case basis and is provided on a paid service basis.

Terms Used

End of Product Lifecycle: A process that contains a series of technical and business milestones and activities, that is associated with the Software Product Management Lifecycle. Once this process is completed, the product becomes obsolete, i.e. the product is not sold, manufactured, repaired, maintained or supported.

End-of-Sale Date: The date when the product is no longer available for sale through Ericsson Sales Channels.

Software Maintenance Support: Ericsson during this timeframe may release software Service Packs or Emergency Bug Fixes to the software products. After this date, Ericsson Catalog Manager Engineering will no longer develop, repair, maintain, or test the software products.

End-of-Life Date: This is a milestone used to indicate that a product is at the end of its lifecycle, and Ericsson Catalog Manager will no longer be marketing, selling or promoting the particular product, product version, product model or service. The end-of-life product may have a replacement product, product version, or product model.

7.3 Documentation

Ericsson Catalog Manager offers a complete set of documentation included with the solution in electronic format.

During the project implementation, there will be a number of project documents to ensure that the requirements have been properly gathered and the appropriate testing and implementation plans are in place.

Also, at the end of the project implementation, Ericsson will deliver the following documents:

- **Functional Specification document.** This document has a high-level functional breakdown of the solution, a description of the user interface, metadata configuration parameters, etc.
- **Systems Administration document.** This document covers things like a description of the software environment that was delivered (e.g. versions



of application servers, operating system, etc.), topology diagrams, and instructions for installing new versions, etc.

The following user guides are also delivered electronically with the product:

- Service Designer Tool User Guide
- Systems Administration Application User Guide
- Resource Editing Tool User guide
- Process Monitor Tool User Guide
- JavaScript Debugger Tool User Guide
- Configuration Tool User Guide
- Installer User Guide
- User Profile Administration Tool User Guide

In addition, a facility exists within the Service Designer to (at any time) auto-generate documentation representing the customer's actual metadata configuration.

8 Software Support Services

8.1 Introduction

Software Support Services include rights to receive all software updates and fixes made Generally Available during the term, along with help desk support for the term of the Support Services Agreement.

Ericsson provides high quality software support via a team of expert support staff. The hours of coverage is determined by the software support level subscribed.

8.2 Scope of Services

Payment of the annual maintenance fees will entitle the customer to the following software support services, in accordance with the level of support ordered:

- Remote support for problem determination, analysis and response;
- Remote maintenance to supply fixes (or work-around or third-party solutions) to solve errors and malfunctions in the software;
- Remote development associated with change requests; and delivery of an electronic copy of all new releases to the software, including fixes, upgrades to the software, new product releases, and any related documentation, which may include release notes and migration scripts.
- The customer is responsible for installation of all new releases to the software.



8.3 Software Warranty

Upon Customer Acceptance of the Licensed Software, Ericsson offers a 90 day warranty period during which software support services are delivered at no additional charge.

8.4 Software Support Services

Ericsson Catalog Manager software support services are divided into 2 levels to suit the needs of the individual customers:

8.4.1 Level 1 – Premium Software Support Services

- Premium support provides an account prime, but no dedicated support personnel.
- Account prime will liaise with the customer on a regular basis, serve as the point of contact for the resolution of issues, and be responsible for keeping the customer informed of new releases and patches as they become available.
- Account prime will work closely with the assigned support personnel to ensure that issues are resolved on time.
- Third party issues, once identified, are coordinated with the parties involved to achieve resolution.
- Support is provided around the clock on a 24x7x365 basis.

8.4.2 Level 2 – Standard Software Support Services

- Same quality of attention as provided in higher levels (levels 1 and 2) but no account prime or dedicated personnel assigned
- Support is provided Monday to Friday, from 9:00 AM to 5:00 PM Eastern time.
- For issues identified as third party related, the Ericsson Catalog Manager team will inform the customer, and coordinate resolution on a time and materials cost basis to the customer, if requested.

8.4.3 On-Demand Services

Additional support required outside the Software Support Services Agreement is available on a time and materials basis, based upon a signed Statement of Work or Change Request.

Interaction with the Customer Service center is handled via a 1-800 number to work directly with a Customer Support Agent to resolve any identified product issues.



8.5 Terms

The following terms are applicable for all Software Support Services offered by Ericsson Catalog Manager:

- Software Support Services are applicable to the Ericsson Catalog Manager licensed software only.
- Application support (i.e. for the Developed Software) is also available at additional fees.
- All Software Support Services contracts are offered on a year by year renewable basis.
- Software Support Services commence upon the expiration of the warranty period.
- Software Support Services fees are payable annually in advance. Support services are renewed each year, unless the customer advises in writing 60 days in advance.



9 Trademarks

Ericsson, the Ericsson logo and the Globemark are trademarks of Ericsson.

Ericsson is a recognized leader in delivering communications capabilities that enhance the human experience, ignite and power global commerce, and secure and protect the world's most critical information. Serving both service provider and enterprise Customers, Ericsson delivers innovative technology solutions encompassing end-to-end broadband, Voice over IP, multimedia services and applications, and wireless broadband designed to help people solve the world's greatest challenges. Ericsson does business in more than 150 countries. For more information, visit Ericsson on the Web at www.Ericsson.com.

10 Disclaimer

This document may contain statements about a number of possible benefits that Ericsson believes may be achieved by working with Ericsson. These might include such things as improved productivity, benefits to end users or cost savings. Obviously, these can only be estimates. Gains might be qualitative and hard to assess or dependent on factors beyond Ericsson's control. Any proposed savings are speculative and may not reflect actual value saved. Statements about future market or industry developments are also speculative.

Statements regarding performance, functionality, or capacity are based on standard operating assumptions, do not constitute warranties as to fitness for a particular purpose, and are subject to change without notice.

This document contains Ericsson's proprietary, confidential information and may not be transmitted, reproduced, disclosed, or used otherwise in whole or in part without the express written authorization of Ericsson.