

Exalt Repository

product description

Implior AB



June 2007

The **Exalt Repository** from Implior AB is a modern document handling system that meets the requirements for document-driven organizations. It solves all problems related to storage, backup, versioning, audit, user authentication and access, distribution, workflow, change management, and retrieval. It is built using modern and proven technology and is subject to a tough quality assurance program.

Simple

Nothing to install - the application is fully web-based.

The very simple user interface virtually eliminates the need for time-consuming training and minimizes the risk for user errors.

Fast

The response time is optimized by using state-of-the art dynamic pages and a well thought-out protocol design.

Secure

Exalt Repository secures your documents from *loss*, *accidental changes* and *unauthorized access*. There is also a mechanism to prevent colliding updates.

Authentication: all access requires user login.

Secure protocol: the portal uses strong encryption with the HTTPS protocol.

Access control based on permissions for roles (groups and/or individuals) ensures that only authorized persons can see or change a document. The simple and clean design prevents administrative errors.

Full audit trail; shows *who* did *what* to a certain document and *when*. All versions are kept.

Locking checkout and checkin prevents colliding changes.

Links: each document has a unique link (URL) that can e.g. be mailed. The receiver must *log in* to the secure portal to see or download the document. This eliminates the need to send a document via unsecure mail.

Backup: Very simple internal design for backup ensures fail-safe database backups.

Powerful

Full version handling; all versions are kept forever. This allows the Exalt Repository to work as a *time machine*; you can look at the documents at a certain date. You can also find all changes between two dates or since a certain date.

Metadata: each document can be tagged with user-defined metadata. The metadata can be configured to be free text or predefined values.

Quick find: There are several ways to find the right documents. You can **browse** directly in the tree-organized folders, or **search** by name, version or metadata.

Workflows are supported using the highly flexible metadata facility.

Packaging: a catalog of documents can be bundled into ZIP files that can be exported and imported.

Mail notification: each user can *subscribe* to changes to documents of interest. Mails can be send immediately or bundled into a daily or weekly digest.

Signatures: a document version can be *signed* by a number of users, indicating e.g. approval.

Wordwide

You can place the server in a secure location and still work from **anywhere** in the world. This is increasingly important in the emerging global organization.

Document names and metadata can be in **any language**; Exalt Repository fully supports Unicode (UTF-8).

Date and time is indepenent of **time zones**. Timestamps are stored in "net time" (UTC) and displayed in local time of the user.

Workgroups

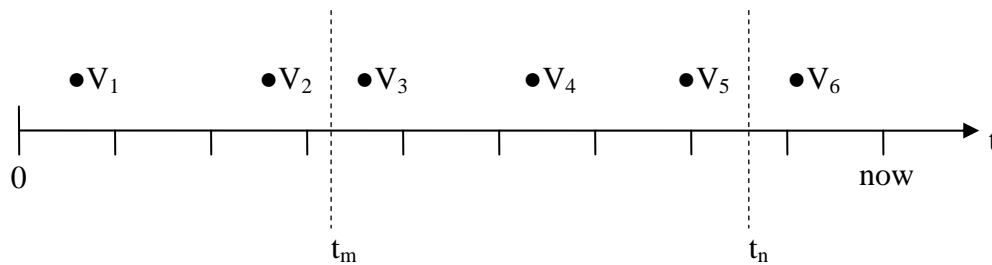
Users can be made members of workgroups, where documents are stored. Thus each project can have its own workgroup. This is very practical in large organizations.

A workgroup is completely invisible to non-members, and only the members can see or download the documents. Thus each workgroup works as an independent document management system.

Future

Planned future extensions includes fully integrated BPM (Business Process Modelling) using the Implior Exalt BPM engine. Complex business rules will be supported by advanced yet simple predicate logic expressions.

More about versions



The version handling is complete and flexible. The user can select any kind of view:

- **single version** - latest before t_a
 1. $a = \text{now}$: current version (V_6)
 2. $a = n$: at time t_n (V_5)
- **multiple versions** - between $t_a - t_b$
 1. $a = 0, b = \text{now}$: all versions (V_1-V_6)
 2. $a = m, b = \text{now}$: new versions since t_m (V_3-V_6)
 3. $a = m, b = n$: new versions between $t_m - t_n$ (V_3-V_4)