T2S General Technical Specifications

General Technical Design

2.2.0

Author	4CB
Version	2.2.0
Date	23/11/2009

1. Introduction and executive summary

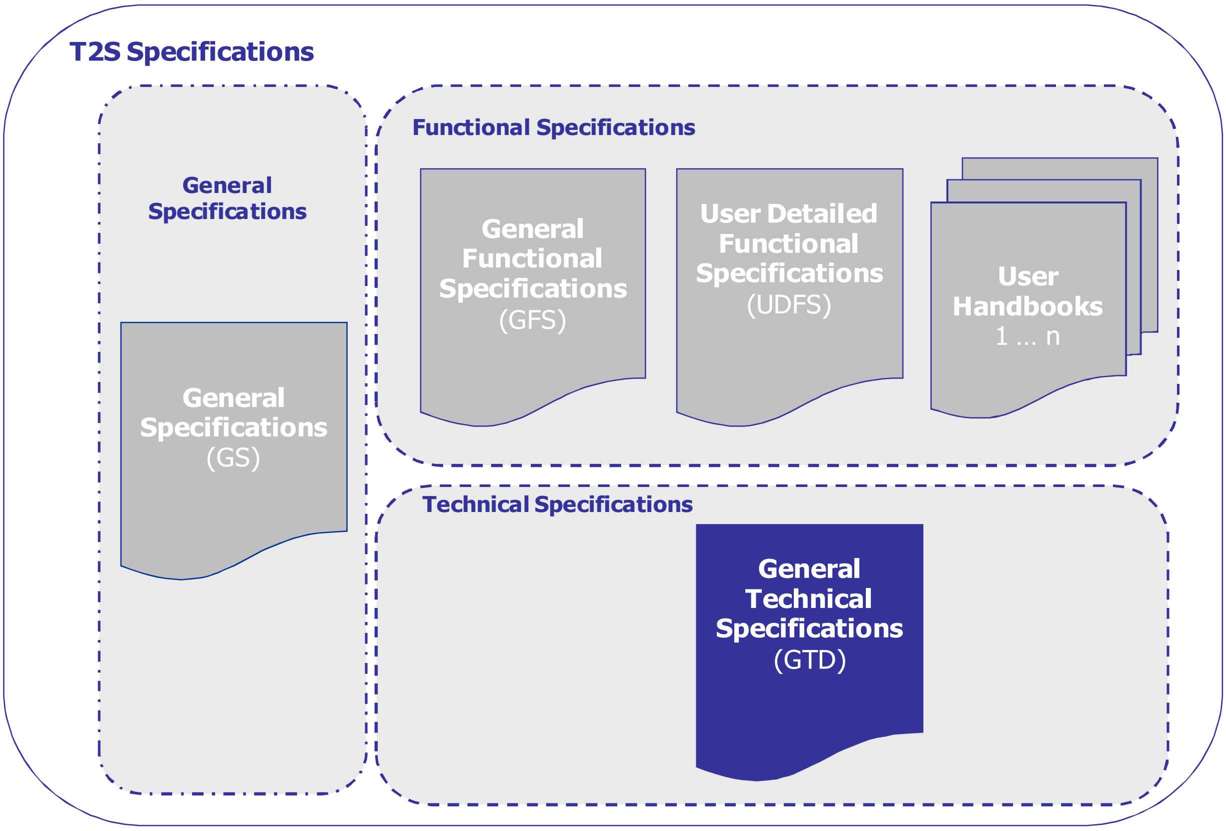
On 8 March 2007, the Governing Council decided that "the T2S service will be developed internally within the Eurosystem and operated on the TARGET2 platform in order to exploit synergies with TARGET2 to the fullest extent". This statement explicitly acknowledged the close relationship between TARGET2 and T2S, and formed the basis for the now commonly used "T2S on T2" concept **{T2S.19.010}.**

T2 and T2S are however two distinct services on one single platform (the Single Shared Platform or SSP): TARGET2 for large-value euro payments and T2S for securities settlement. This solution allows exploiting synergies between the two infrastructures, while avoiding tight and risky dependencies between critical services.

1.1. Overview of the T2S deliverables for the specification phase

The T2S deliverables for the specification phase are considered as documents aiming at allowing users to understand how services described in the T2S User Requirements Documentation will be provided by the T2S platform.

The diagram below presents an overview of all the T2S deliverables for the specification phase.


T2S Specifications
General Specifications
General Specifications (GS)
Functional Specifications
General Functional Specifications (GFS)
User Detailed Functional Specifications (UDFS)
User Handbooks 1 … n
Technical Specifications
General Technical Specifications (GTD)

1.2. Overview of the 4CB solution

T2S will be a very critical system for the European financial industry and will be undoubtedly a highly demanding application, which will require a state-of the art technical infrastructure:

- The T2S architecture offers very high levels of performances in order to process smoothly the volumes in any circumstance, especially on the peak days;
- The T2S architecture is easily scalable to adapt the processing capacity to the increasing volumes of transactions;
- The T2S service is highly available and operated uninterrupted 22 hours per day
- During day time response times stays within a short range (low deviation), even on the peak days;
- The service demonstrates an optimal capacity of settlement based during both day time and night time;
- The service offers a state of the art user interface for user to application mode as well as for application to application mode.

Perfectly aware of these outstanding requirements of T2S, the 4CB took advantage of their expertise and their experience, and benefited from the support of their technical providers for the preparation of the design of the infrastructure and the application architecture. The present design is based on the most advanced technology for the interface together with a back end application implemented on technical infrastructure with proven reliability, scalability and security. This architecture already proved the ability to process huge volumes of transactions in T2.

The interface infrastructure is based on a state of the art multi-tier architecture which has already proven to be able to process seamlessly both U2A and A2A communications. The architecture is natively designed to process XML messages and follows a SOA approach. To deal with the expected high number of XML messages (managing XML messages is an highly processor bound activity) the architecture is furthermore enhanced with special technologies like dedicated XML appliances. The high flexibility is guaranteed by the usage of an enterprise service bus in conjunction with a widely used Application Server (IBM® WebSphere®). This technology, based on the MVC (Model-View-Controller) architecture, ensures the separation between the presentation layer and the business logic. The architecture includes authentication and identification mechanisms based of the most up to date technology.

Processing of business transactions is implemented on the IBM System z mainframe platform which is able to provide resilience and highest availability. The two regions/four sites principle guarantees the most up to date answer to prevent systemic risk in every circumstance. The System z is designed to ensure application availability up to 99.999% which equates to approximately 5 minutes of downtime in a year on average. The Parallel Sysplex® technology enables several IBM z/OS® systems to behave as a single, logical computing facility for continuous availability, high performance and no single point of failure. Maintenance operations can be conducted on one partition without any impact on the operation. Thanks to the Resource Access Control Facility the System z is extremely secure and protects vital resources and control access.

Parallel Sysplex architecture, together with capacity on demand, guarantees the flexibility to adapt the power of the system to high peaks of transactions. The DB2 data base the IMS transaction manager and Cobol2 are commonly used on this infrastructure in numerous sites in the world and most specifically in the financial industry.

T2S architecture provides a valid answer to the technical challenges of an European-wide settlement system offering a fully scalable central processing system, a storage sub-system with synchronous and asynchronous mirroring and a dedicated network connecting the processing sites (4CBNet).

The present General Technical Design defines the high level technical architecture for T2S based on currently available technology. By 2013, any technical advancement could lead to further investigation and possibly changes in the described architecture.

Concerning the IT and operational management aspects, the 4CB organization, largely built upon TARGET2, constitutes a solid basis for the intensive service levels expected from a platform as critical to business as T2S

1.3. Business and technical assumptions

The application described in Chapter 2 and the infrastructure described in Chapter 3 have been designed and scaled based on the assumptions hereafter:

Business continuity concept

The full integration of T2S in the SSP architecture allows the new system to inherit from TARGET2 a state-of-the-art business continuity model, capable of handling short continuity failure, major failure or local disaster (intra-region recovery) as well as regional disaster (inter-region recovery).

Volumetric assumptions

ECB on September 2009 provided volume estimations based on the following assumptions:

- T2S is expected to be launched in 2013 and the migration period to last until mid-2014. The year 2014 will have the full production of T2S operation so it is used as reference instead of 2013.

- 2008 and 2009 data were extracted from the Blue Book (as was the case for the URD for the year 2007), the basis for the new analysis is to be the total of 2008 settled volumes of all euro-zone CSDs plus total of 2008 settled volumes (euro and local currency) of UK, Denmark, Sweden, Estonia, Latvia, Lithuania and Romania. The data for Sweden and Romania are still in verification with the local CSDs. The data for Estonia, Latvia, Lithuania, Romania, Slovenia and Slovakia are taken from the ECB Blue Book. No data are available for Iceland.

- The volume for 2009 is to be the 2008 total market volume minus 15%, in line with currently observed market volume developments. It is to include additional volume impact of Sweden, Denmark and Finland at the end of 2009 to account for the participation of the Nordic CCP.

- Market growth factor of 5% is to be applied as from 2010 volumes onward.

Due to these assumptions, volume estimates for T2S are the following:

YEAR	ANNUAL VOLUME OF TRANSACTIONS	AVERAGE DAILY VOLUME[1]
2008	260 524 500	1 009 785
2009	227 868 000	883 209
2010	221 279 000	857 671
2011	232 342 950	900 554
2012	243 960 098	945 582
2013	256 158 102	992 861
2014	**268 966 007**	**1 042 504**
2015	282 414 308	1 094 629
2016	296 535 023	1 149 361
2017	311 361 774	1 206 829
2018	326 929 863	1 267 170
2019	343 276 356	1 330 529
2020	180 220 000	1 397 054

Table 1-2 New volume estimates

[1] Average daily volume = Annual Volume of Transactions divided by 258 operating days in a year. Only half year (6 months) has been considered for 2010.

The following figures show the estimated volumes to be managed by T2S core system (the mainframe) and the archiving platform in 2014 first year of production.

Definition,	**Volume**	**Comments**
Annual volume of transactions	**268 966 007**	
Average daily volume	1 042 504	Average daily volume = Annual Volume of Transactions divided by 258 operating days in a year
Average night time volume	938 254	Night time volume is estimated to be 90% of the daily total (Average night time volume and average day time volume have an embedded margin of 20%)
Average day time volume	312 751	Day time volume is estimated to be 30% of the daily total. (Average night time volume and average day time volume have an embedded margin of 20%)
Peak day workload	4 326 391	Peak day workload is calculated as the average daily volume multiplied by a peak load factor which is provided in most markets by the CSDs.
Peak night time work load	3 893 752	
Peak day time work load	1 297 917	
Night time peak hour work load (10h/night)	389 375	
Day time peak hour work load (12h/day)	108 160	

Table 1-3 Estimated transactions volume in the first year after the migration period

The figures have been used to review the initial estimation of the technical resource needs (processing power, storage configuration etc.) for the various T2S environments. From a technical point of view, T2S infrastructure and applications are scalable to deal with in the future requested changes in the above listed figures.

Availability requirements

The planned level of availability defined by the URD for T2S is above 99.7%, calculated on annual basis **{T2S.20.320}**. The ability of T2S to satisfy this requirement will be assessed in the context of the business continuity tests as well as other resiliency tests.

According to the Information Technology Infrastructure Library (ITIL®), availability is "the ability of a configuration item or IT service to perform its agreed function when required. Reliability, maintainability, serviceability, performance, and security determine availability. The calculation of availability is usually on a percentage basis with reference to an agreed service time and downtime. It is best practice to calculate availability using measurements of the business output of the IT Service."

Concerning Availability Management, Chapter 20 of T2S User Requirements Document (URD) states that "The goal of the Availability Management process is to optimise the capability of the IT

Infrastructure, services and supporting organisation to deliver a cost effective and sustained level of Availability that enables the business to satisfy its business objectives."

In compliance with ITIL® standards, the main Availability criteria that T2S has to fulfil can be classified as described hereafter:

- Reliability;
- Maintainability;
- Serviceability;
- Performance;
- Security.

Performance requirements

T2S is able to handle the estimated settlement volume running real-time settlement in parallel to a continuous optimisation algorithm without degradation of service level **{T2S.17.030} {T2S.17.040}**. This is ensured by an application design which allows a scaling via parallel processing as well as an infrastructure design which allows a parallelisation of the underlying components.

Furthermore, T2S does not have any performance impact on TARGET2 activities, and vice versa **{T2S.17.050}.**

In compliance with the response time requirements defined in the URD,

- T2S responds to 95% of the basic queries[2] User-to-Application (U2A) or Application-to-Application (A2A) mode within 3 seconds **{T2S.17.140}**;
- Any data to be created, modified, deleted via the user to application interface is updated in real time **{T2S.17.160}**.

Business monitoring tools will check that T2S is compliant with the Service Level Agreement (SLA) and produce corresponding reports.

Performances and throughput will be assessed in the context of the stress test campaign.

[2] Queries to retrieve a single object (status of one instruction, static data for one ISIN etc.) Any other query can be considered as a complex query.

2. Application Design

2.1. General overview

2.1.1. Architecture overview

As stated in the initial proposal, the T2S system makes use of the infrastructural services currently available for TARGET2 **{T2S.19.010}**.

Therefore the T2S "core business" applications (e.g. Life Cycle Management and Settlement) are designed to run on the mainframe (z/OS), using IMS/DC as a transaction manager and DB2 as the relational database management system. This choice of proven and widely used technologies allows T2S taking advantage of the benefits of a highly reliable, robust and secure technical architecture. Message Queues are used for the implementation of asynchronous T2S-internal communication. The Legal Archiving facility is a separate environment backed by Centera and Windows servers.

The T2S "front-end application" (U2A interface) is based on the Java Enterprise Edition platform using an IBM WebSphere application server. XML facilities for message parsing, transformation and routing will handle the A2A traffic.

The combination of mainframe technologies for the "back-end" application and of open system based technologies for the "front-end" takes the best advantages of both technologies:

- reliability, robustness, scalability, efficiency and security from the mainframe technologies,
- modern state-of-the art and user-friendly graphical and XML based interfaces from the open system technologies

which guarantees a flexible and modular design. In addition, the experience of the 4CB in these technologies will allow limiting the risk related to the implementation of the T2S technical infrastructure to a minimum.

The following diagram gives an overview of the application architecture used for the implementation of the technical domains. The details of the underlying infrastructure are described in chapter 3 "Infrastructure Design".

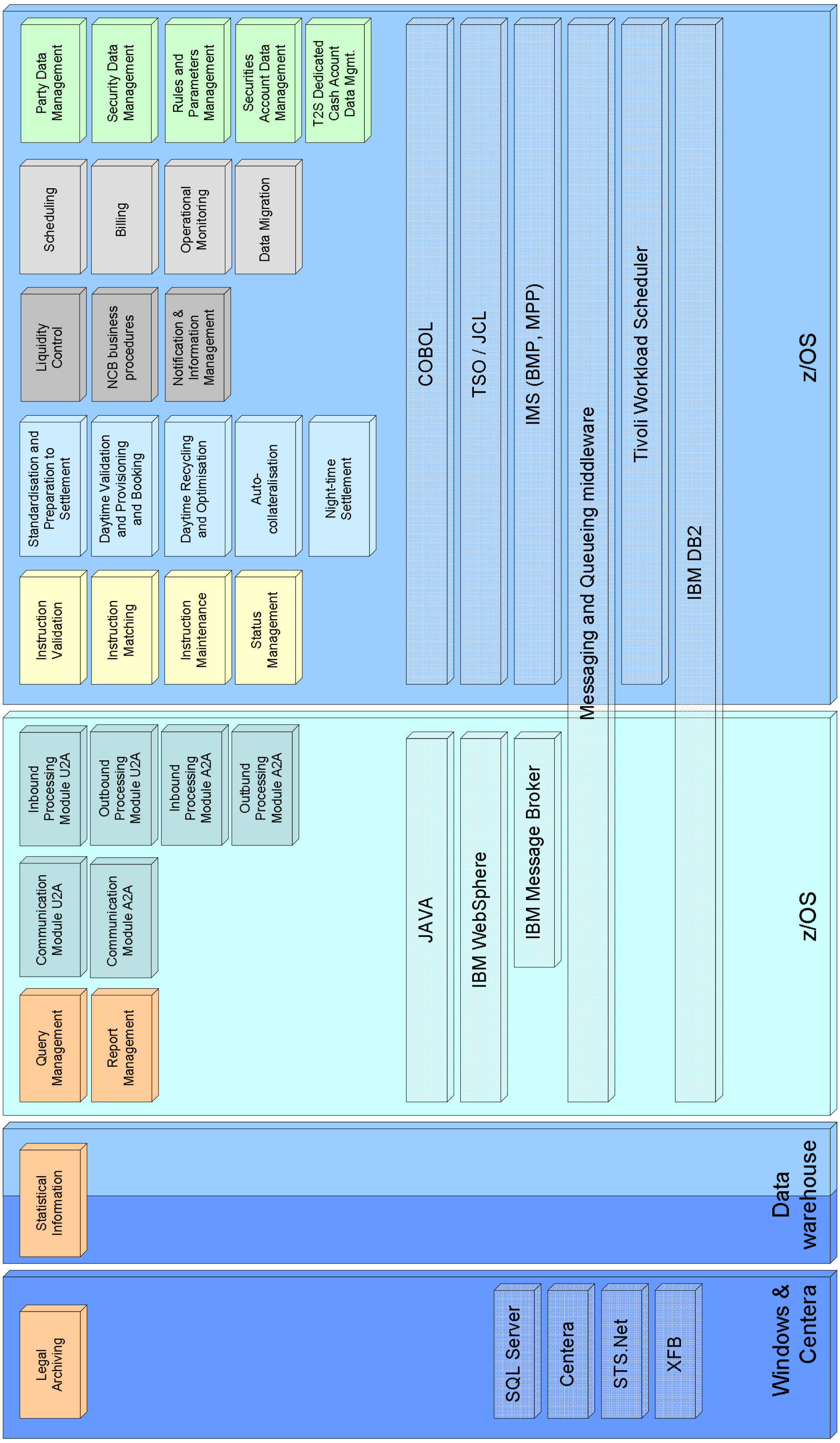


Figure 1 – Technology and middleware layers

The following diagram depicts a high-level view of the T2S system, the services the system provides, the users and how the users access the system.

Communication with the end users and other IT systems is done via dedicated networks and network providers using standard communication technologies and protocols which adhere to the ISO 20022 XML standard. **{T2S.19.230}.**

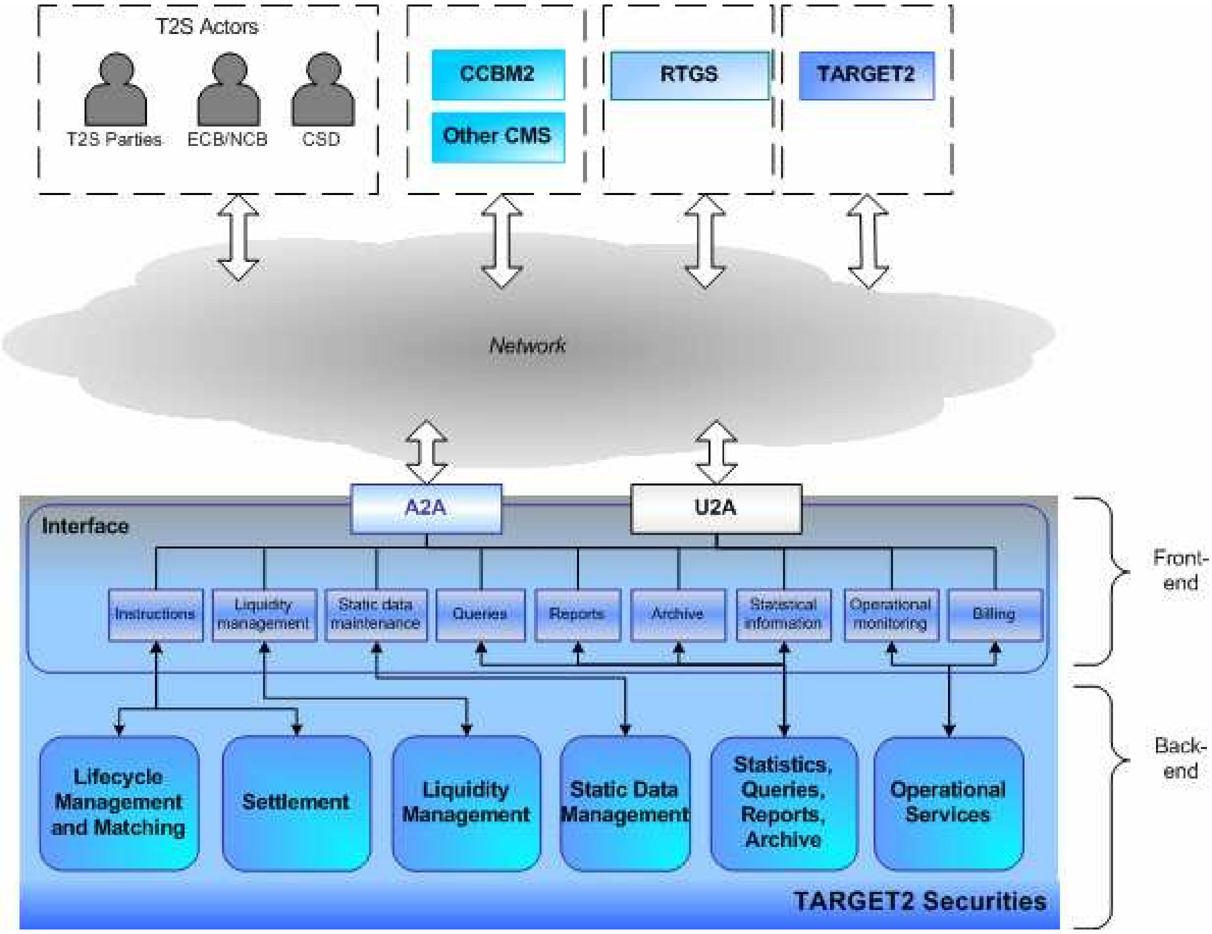


Figure 2– High level overview and external connectivity

A more detailed and **technical view** showing the internals of the T2S system is depicted on the following side.

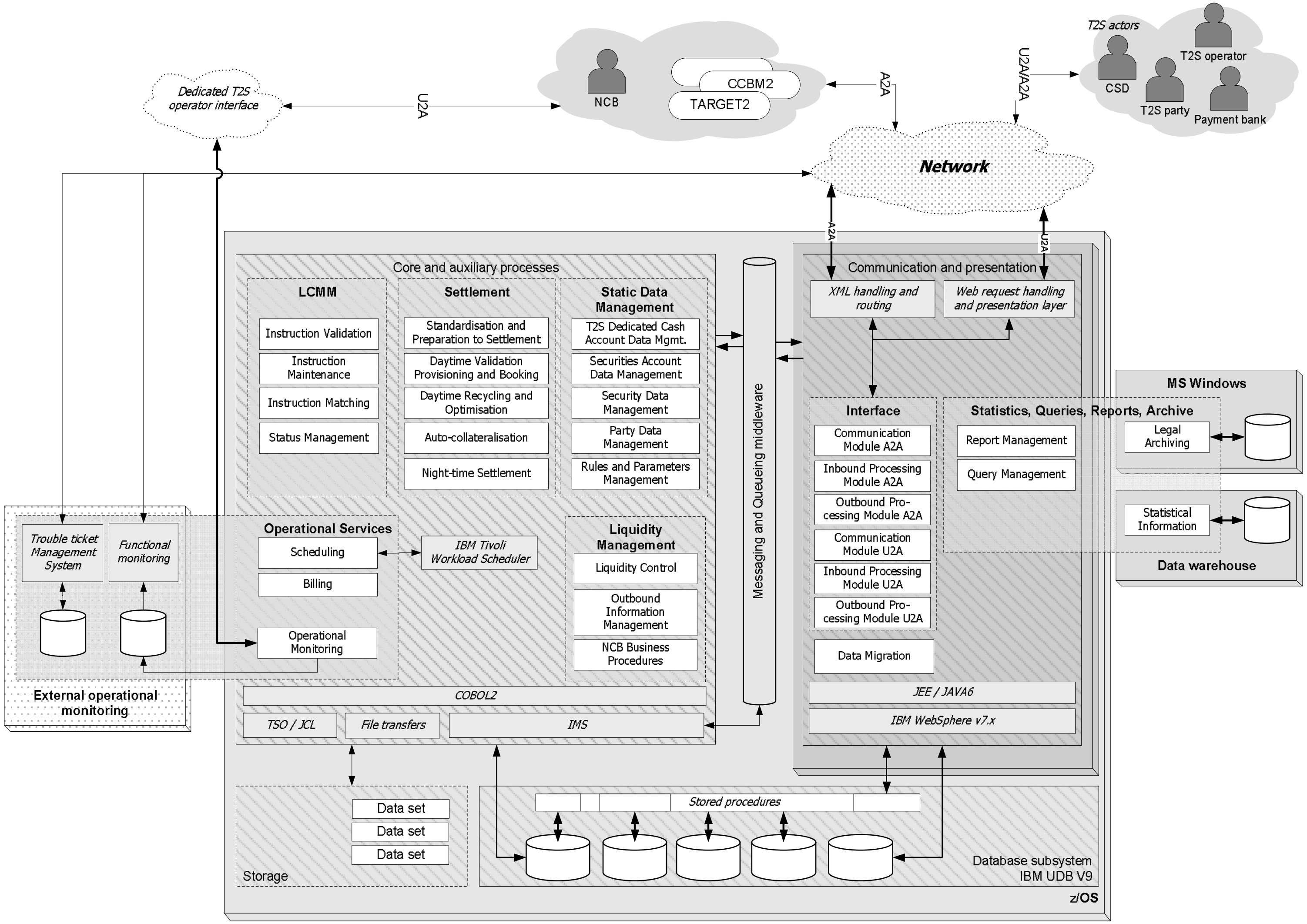


Figure 3 – Technical view

2.1.2. General design descriptions

2.1.2.1. Components of the application

The T2S application is comprised of a set of modular and independent technical components which relate to the functional domains and modules found in the GFS. Each component is dedicated to the fulfilment of a specific task. The actual choice of technology used for the implementation (i.e. the programming languages) and the hosting (i.e. the runtime environments) of the components is based on the functional and non-functional user requirements and follows a best-fit approach. Generally the system is designed as a multi-tier architecture in which the presentation layer, the business logic layer and the data layer are clearly separated. **{T2S.19.170} {T2S.19.240}**.

All components which make up the T2S system are loosely coupled and are capable of acting independently from each other. Each component offers functions to other components via dedicated technical interfaces and hides its internal data representation and implementation details behind a so called facade. The implementation of the internal technical interfaces is based on the usage of the communication mechanisms provided by the middleware (i.e. IMS and WebSphere MQ for messages). **{T2S.19.130} - {T2S.19.140}.**

The adoption of a modular approach combined with the principle of loosely coupling between the modules will facilitate the development of the different functionalities within the development phase, simplify the management of the complexity in T2S and eases the maintenance of the application once T2S is live. Additionally this approach allows the application to be more error-tolerant because even if one module fails or abnormally terminates ("abend") other modules can continue their work while loss of data is avoided by the mechanisms of the used middleware.

It has to be emphasized that in order to reach the requirements in terms of throughput the internal communication between the modules is not based on an enterprise service bus but on the above described communication mechanism. SOA standards like enterprise service bus implementations will be supported only for the communication with external systems (i.e. within the interface domain).

2.1.2.2. Database design and storage

In general, the design of the data model will follow a relational approach. In order to allow a traceability of the technical data model with the functional data model adequate naming conventions will be utilised (described in a separate technical document) as well as appropriate tools for a round-trip-engineering technique.

As the T2S system is required to be capable of dealing with a large amount of data and the efficient and effective processing of very high numbers of transactions, special attention must be paid to the design of the technical database model and the storage of data. In particular, all efforts put into an optimised implementation provide a great benefit for the most critical parts of the whole application (e.g. the settlement engine and the optimisation algorithms).

The design of the database on both a logical and physical level takes into account the needs to guarantee performance consistent with the SLA and the goals of the service provisioning. The technical design of the database relies on a central repository for the system data (meta data) and is optimised to deal with data accessed in real-time and data used in a less frequent way **{T2S.19.160}**.

The data model and database design ensures a separation of static and transactional data by using system entity identifiers. **{T2S.11.080} {T2S.19.190}**. The system entity identifiers find their representation in the data model in a specific column.

Each domain maintains a dedicated part of the technical data model and the physical database according to the business processes belonging to this domain. Data residing in a different domain can be accessed via dedicated technical interfaces. The data integrity is ensured by the functional design of the application (data ownership: eg. SD data can only be changed by SD domain).

Additionally, following concepts and techniques are used during the design of the database:

- partitioning of tables and table spaces;
- creation of indexes and additional technical columns;
- union of more than one conceptual entities in the same physical table;
- division of one conceptual entity in more than one physical tables.

Those techniques are used for performance improvements during the physical Database design.

2.1.2.3. Internal technical Interfaces

The following standard technologies have been identified for the implementation of the **internal technical interfaces {T2S.19.150}:**

- In the T2S system, **Message Queues** are used to implement technical interfaces providing means of asynchronous communication
- **IMS message queues** are used to implement technical interfaces between technical modules;

For information regarding the interface technologies used for the implementation of external technical interfaces please refer to the chapter 2.2.1 ”Interface”.

2.1.2.4. General architecture

Generally the T2S system is divided into a front-end part and a back-end part (see 2.1.1 “Architecture overview”).

Front-end

The Front-end part is based on the Java EE (Java Enterprise Edition) technology and runs on the Java EE server of IBM, the WebSphere Application Server (WAS).

Java EE is designed to support applications that implement enterprise services. The Java EE application model defines an architecture for implementing services as multi-tier applications that deliver the scalability, accessibility, and manageability needed by enterprise-level applications. This model partitions the work needed to implement a multi-tier service into two parts: the business and presentation logic to be implemented by the developer, and the standard system services provided by the Java EE platform.

The IBM WebSphere application server delivers the secure, scalable, resilient application infrastructure for JEE applications by providing web and Enterprise Java Beans (EJB) containers. Such a container enables the relevant java components to be executed using a variety of generic services, which are provided by the container. There is no need to develop these services inside the application. Examples of these services are Persistence, Transaction Management, Security, Scalability and Management. The Java EE platform also provides the XML APIs and tools needed to run web services.

Java EE applications are made up of components. A Java EE component is a self-contained functional software unit that is assembled into a Java EE application with its related classes and files and which communicates with other components.

Containers are the interface between a component and the low-level platform-specific functionality that supports the component. Before a web or enterprise bean component can be executed, it must be assembled into a Java EE module and deployed into its container. The assembly process involves specifying container settings for each component in the Java EE application and for the Java EE application itself.

The web container (belonging to the web tier) hosts technologies to handle the presentation logic and manages the execution of JSP page and servlet components for Java EE applications. The EJB container is responsible to support the business logic and manages the execution of enterprise beans for Java EE applications.

All business code (logic that solves or meets the needs of a particular business domain) is handled by enterprise beans (EJB) running in the business tier.

An overview of a typical JEE multi-tier architecture is shown in the picture below.

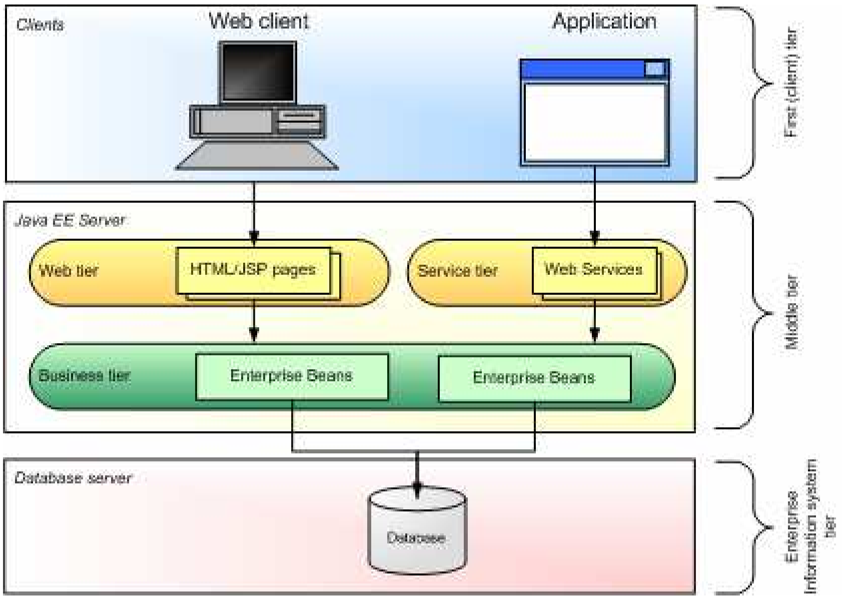


Figure 4 – JEE multi-tier application architecture

Back-end

The main type of IMS applications used is MPP-based processes (Message Processing Programs). This allows a real-time parallel processing in a predefined number of IMS regions (enabling scalability).

In a parallel system, the applications, which can be compared to threads, are designed as short processes.

The design of the applications takes into account the principles of synchronous and asynchronous operations. The processes are required to perform within their own "unit of work", simply using the synchronised functions specified or required to keep integrity of the information. All functions which can be done in an asynchronous way are implemented in different processes.

Most processes are able to run in parallel (e.g. Instruction Validation in the LCMM domain) in order to enable horizontal scalability. The impact of concurrent accesses to the same information is minimised to avoid undesirable coupling among processes. As a general rule, the status of the system is maintained exclusively in the database, this means all data is kept in the database and never in any transient state like in-memory. Any Logic of a single module (e.g. Instructions Validation) executes one or more transactions in that database (i.e. a unit-of-work does not span more than one module).

Therefore scalability happens at the module level and is configured externally to the modules. The logic of a module can be split in multiple program units, running in multiple environments:

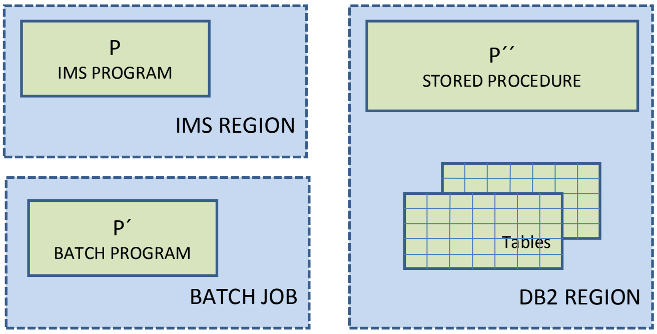


Figure 5 – Processing regions

Activation of on-line individual instruction processes (e.g. validation of a particular incoming instruction) takes place as soon as such instruction arrives and there is processing capability available, in order to avoid any delay. Moreover, the activation can physically happen in any IMS or DB2 region configured for that particular process.

In the following diagram, a generic program unit responsible for certain functionality is depicted as a small rectangle (P, P', P'') whereas multiple parallel rectangles indicate the possible parallel existence of more than one instance of the corresponding entity.

The parallel sysplex environment architecture provides the support for the parallel processing and data sharing among different z/OS systems and subsystems. **{T2S.19.180} {T2S.17.040}**.

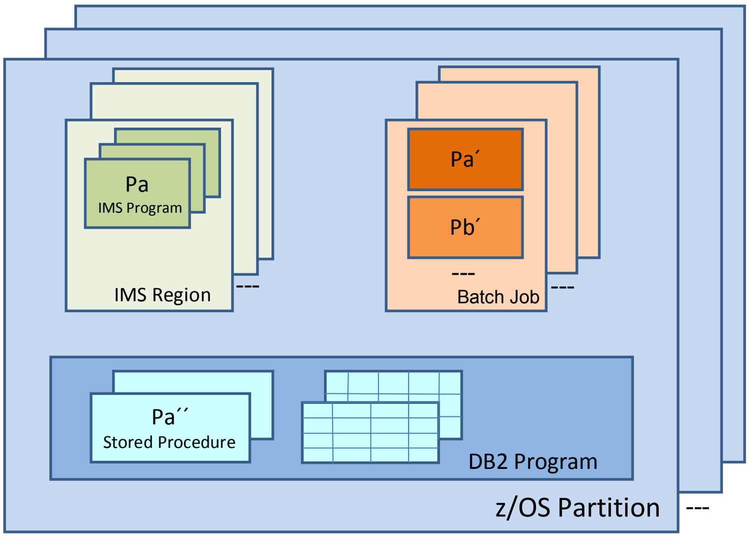


Figure 6 – z/OS partition setup

2.1.2.5. Application security

Application security relates to the tasks of **authorisation** and **authentication**. The necessary checks are implemented by T2S (e.g. Role-related checks)

Authentication: checking the user is the one he/she claims to be (typically by validation of userid / password and/or certificates). In case the end user connects to T2S via a network provider which does not offer authentication services, this check is performed by the T2S system.

The **authentication** and the related certificate handling occur on the infrastructure level. Additional layers of security like the encryption of messages and communication channels are also handled on infrastructure level.

Authorisation: checking if the user has the rights (i.e. the roles) to perform a given task in the system. A role in the application security context relates to one or more specific pieces of functionality found in an IT system. A user of an IT system can be assigned none, one or more roles. Hence the roles assigned to an user govern what functions a user actually is allowed to use in an IT system.

T2S implements a role-based access control (RBAC) mechanism to ensure that every access to the system and the invocation of a function can only be performed by authorised users. These checks are performed independently of the network provider chosen by the end user. This type of check is called a **role-related** check. A potential example is a role which allows the user to access the function "create account" in T2S. The T2S system allows authorised end users to define their own user roles.

Additionally a second type of authorisation check occurs in the T2S system which is based on a specific set of data adding another layer to further regulate and restrict functionalities a user can invoke. This type of checks is called a **data-related** check. A potential example is an order limit (i.e. the user is only allowed to enter instructions up to a limit of EUR 1 M), or the user is only allowed to work on a specific subset of accounts.

Role-related checks are performed by the Interface domain whereas **data-related** checks are performed by the back-end domains which are responsible for the data involved. Exceptions from these rules, either for performance and/or usability reasons, are possible and in such cases the data-related checks are performed in the Interface domain, too.

The T2S system always needs to conduct role-related checks and optionally (depending on the functionality requested) data-related checks. A data-related authorisation check always requires a role-related authorisation check. Both work in conjunction. The opposite is not true, i.e. a role-related authorisation check does not necessarily require a data-related authorisation check.

2.2. High level design of the modules

2.2.1. Interface

2.2.1.1. Architecture overview

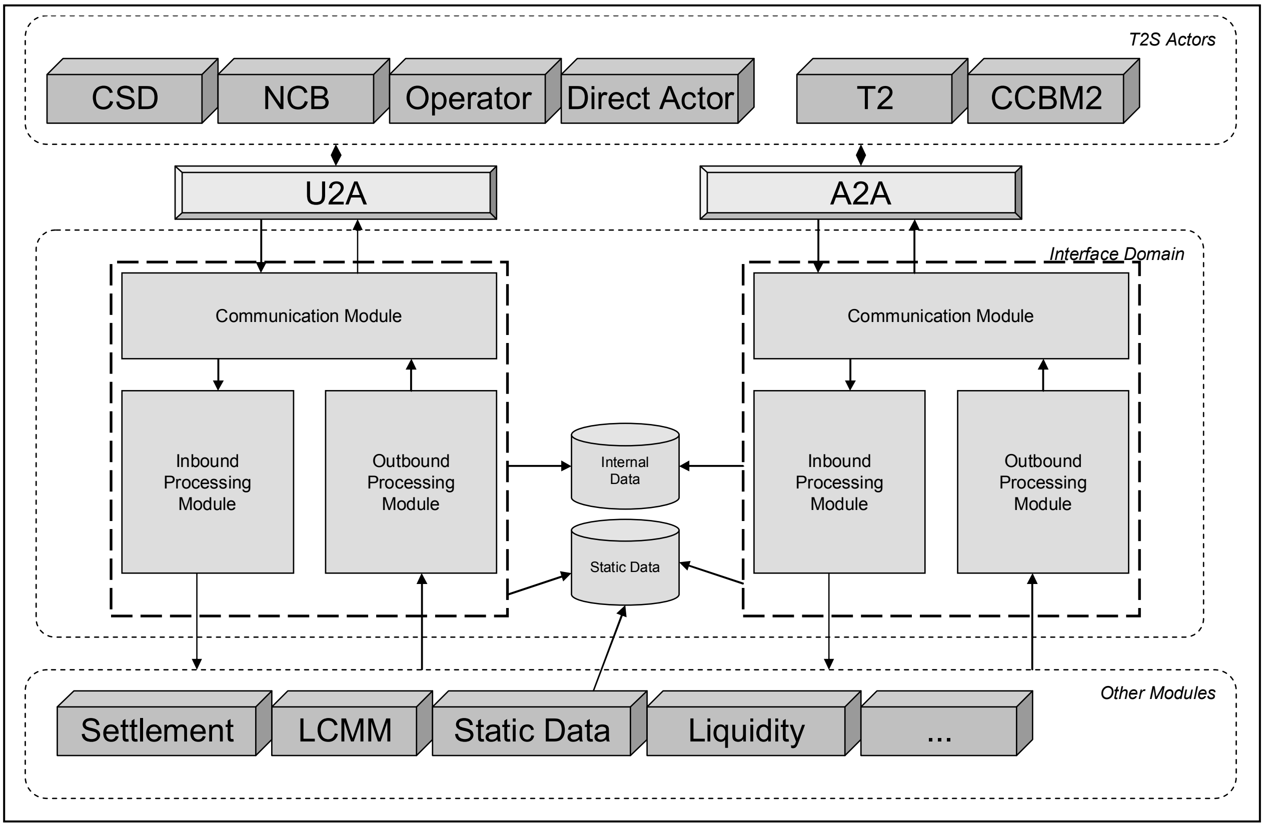


Figure 7 – Interface architecture overview

The Interface Domain handles all incoming and outgoing communication (except some parts of the Operational Monitoring) with all T2S actors and deals with all the networks and communication channels used **{T2S.12.060}**. Also all externally used formats, especially XML as ISO 20022 compliant and proprietary messages and files are processed here **{T2S.19.230}**. The Interface Domain is the single entry and exit point of the complete system (see exceptions above) and therefore hides all tasks related to the external communication from the other domains.

Another job of the Interface Domain is to ensure the role based parts of the authorisation. Also other generic features like syntax checks, navigation and routing to the backend modules are centralised and provided by the Interface Domain.

2.2.1.2. Processing principles

The Interface Domain is presenting the external communication to other ESCB systems uses standard message formats **{T2S.12.340} { T2S.12.350} {T2S.12.360}**, means it will communicate with all RTGS and Collateral Management systems only using standardized ISO 20022 XML messages and files. No other communication formats e.g. to prefer some external systems will be invented. The Interface Domain will support the Eurosystem Single Interface concept **{T2S.12.230} {T2S.12.240}**, e.g. it will follow the ESI style and navigation guide.

As it eases the work to create a Web application for the U2A approach as well as to process a large number of XML data for the A2A approach the Interface Domain is implemented using the Java Technology running on IBM WebSphere Application Server (WAS) and WebSphere Message Broker. Additionally state-of-the-art technologies like enterprise java beans (EJB), message-driven beans (MDB), cascading style sheets (CSS) and asynchronous java and XML (AJAX) are used.

There are high requirements on the performance, throughput, scalability and availability of the Interface Domain, including availability during the Maintenance Window. Therefore multiple and identical instances of the Interface Domain are running on a WebSphere Cluster, thus each instance can and will handle the complete scenario on its own. That means the each part of the complete functionality of the Interface Domain is highly scalable. Additionally the Cluster guarantees a better work balance and failure safety. These instances are synchronised among each other using the Message Queue (MQ) publish/subscribe mechanism.

The two approaches for external communication U2A and A2A are strictly divided in the Interface Domain internally (even if they are related to the same functionality) to avoid bottlenecks and reach the given time restrictions.**{T2S.17.140}**. This is guaranteed as the U2A track is running on the Websphere Application server while the A2A track on WebSphere Message Broker. When it reaches the internal interfaces towards the Backend Modules, the diversification ends, means both approaches use identical interfaces to the Backend Modules as far as possible (i.e. for some special A2A requirements e.g. for bulky data transfers dedicated interface designs will be envisaged). Anyhow there is no knowledge in the Backend Modules if the request was done via U2A or A2A. That simplifies the implementation of their functionality..

Communication Module

Inside the Communication Module the de- and encryption, the certificate handling including authentication and all network related and network dependent tasks are placed. All that is implemented as infrastructural tasks mainly based on third party products. More information therefore will be provided in chapter 3.

Additionally the Outbound Queuing depending on Scheduler events, the T2S Actor type and the response type are handled here. This will be done using persistent MQ technology for storing the messages and specialized Message Broker Nodes for the restarting.

Inbound Processing Module

If the input is a file after a File Validation using Schema files and a double input check on base of the file reference the file is split into single messages. All that is done using multiple instances of Message Broker in parallel to enhance the performance. After that the processing is the same as for single messages.

If the input is only a single message first a validation of the input data is done. Completeness, structure and the format are just some examples for that. In the A2A approach the first part of these checks are executed in form of XML schema validations **{T2S.12.070}** using the ISO 20022 XML Schemas. Additional all A2A checks not already covered by the Schema files as well as the checks in the U2A approach are done using self developed Java functions, e.g. if a special field is mandatory only if also another one is filled, or if a parameter needs restricted values during a special day time. If one of the schema or java checks fails the user gets an error response with a meaningful description of the problems.

All role based authorisation checks are done by the Interface Domain. In case of a "negative result" of the authorisation check (i.e. an authorisation failure) means the user does not have the needed privileges, an error message is sent to the participant.

Queuing depending on Scheduler events and the request types are handled here. This will be done using persistent MQ technology for storing the messages and specialized Message Broker Nodes for the restarting.

The Message and Data Routing includes the distribution of the incoming requests to the correct domains and the collection and generation of the responses. Responses and "one way messages" (pushed messages without request) are sent towards the external actors.

It is also possible to resend messages already sent to T2S actors for contingency reasons in case of problems. Therefore special requirements have to be fulfilled to check the permissions of the requestor, if the T2S system user is allowed to resend the required message .

Outbound Processing Module

First the possible Recipient List needs to be defined, means to create a list of parties which are allowed to receive the business data based on Static Data information. After that it needs to be checked within that list whether they opted for receiving the message again based on Static Data information and if so the message is sent to the actor. Therefore in U2A mode just a synchronous response is done, in A2A mode the technical routing address is catched from Static Data.

As in A2A mode the messages have to be sent to the actor in the correct order, a Message Sequencing using a special database table has to be done for certain message types. Whenever one of these massage types has to be sent a look into this table shows if the correct sorting is fulfilled up to this message. If there is an entry missing this message is just stored and sent with the next attempt, when the missing one is available.

U2A

The U2A approach is provided for human users using a modern web browser (following the ESI style and navigation guide) **{T2S.12.230, T2S.12.240}** not requiring a local installation. Therefore a Java based Web application, presenting a Graphical User Interface (GUI) in XHTML format is used. The main target is to support the user with a clear, easy and comfortable to use interface with a consistent look and feel. An own style-guide will be developed, based on the ESI style and navigation guide. Therefore all state-of-the-art technologies needed to reach that goal (e.g. JSF, AJAX, CSS) are combined in order to provide an consistent front-end to the end-user which is based on the English language for all end users **{T2S.19.270} {T2S.19.280}**. Special requirements for barrier-free applications suitable for handicapped people like keyboard based operability will not be supported, as there is no requirement for in the User Requirements Document.

The Interface Domain is designed according to the Model-View-Controller pattern (strict separation of content, optical presentation and controlling). Thus an easy evolution of the design and the adoption of newer technologies is supported. To generate the View (XHTML pages) the Java Server Faces (JSF) and Facelets technology is used. The controller is implemented by Java Servlets and the model is generated by the kernel Interface Domain application.

A2A

The A2A approach is provided for external back-end applications as well as RTGS and Collateral Management systems. Therefore a XML communication format compliant with the ISO 20022 standard **{T2S.12.040} {T2S.12.050}** is used, fulfilling the requirements as described in the Giovannini protocols.

In this part of the Interface Domain the complete XML part is handled means file splitting, schema validation and transformations. The communication to the Backend Modules is done using internal formats to ease the process, save resources and enhance the performance of the system, means the Backend Modules do not have to handle XML.

A very important part of the A2A approach is the insertion of settlement and maintenance instructions which has very high requirements to performance and throughput. This is reached by the usage of standard, largely scalable broker software (WebSphere Message Broker running in a cluster configuration), especially in order to deal with performance critical parts (XML schema validation and transformation).This broker system is designed to guarantee a well performing, stable system. Additionally the Interface Domain is designed to let all flows and functionality completely run in parallel. Means each request can be handled by multiple instances of the Interface domain. Thus a high failure safety is guaranteed and the system is easily scalable by simply adding more instances.

2.2.1.3. Data

Relevant static data, especially used for authorisation checks, e.g. Static Data information on users and privileges is stored in an own cache in the heap of the WebSphere for performance reasons. This cache will be updated at the "End of Day" as well as for Static Data Intraday Updates after the triggering from the Static Data Domain.

All information relevant to ease the usage of the system for a human user is stored inside a HTTP Session related to this specific U2A user (to avoid a re-login of the user for each action) This includes preferences, state of the art patterns like the "breadcrumb pattern" (a navigation technique to give users a way to keep track of their location within the application, e.g. Home page → Section page → Subsection page) and business information of certain business workflows as well as login, navigation and identity information.

Finally, there are some business information owned and stored by the Interface Domain, e.g. the payload of the original XML messages and audit logs. Most of this data has to be archived in the Legal Archiving module.

2.2.1.4. A2A Messages

All A2A messages and files exchanged between the participants and the Interface Domain are compliant with the ISO 20022 standard. All A2A XML messages are completely processed, validated and finally transformed to internal formats by the Interface Domain. This internal format is then handed over to the Backend Modules. Thus, the Backend Modules will not work on XML but on optimized internal formats.

2.2.2. Static Data Management

2.2.2.1. Architecture overview

The following diagram represents the software objects implementing the Static Data Management domain functionalities, the database and the main internal and external data flows. Of course the software objects are identified starting from the functionalities, nevertheless there is not a 1 to 1 relationship between functions and software objects.

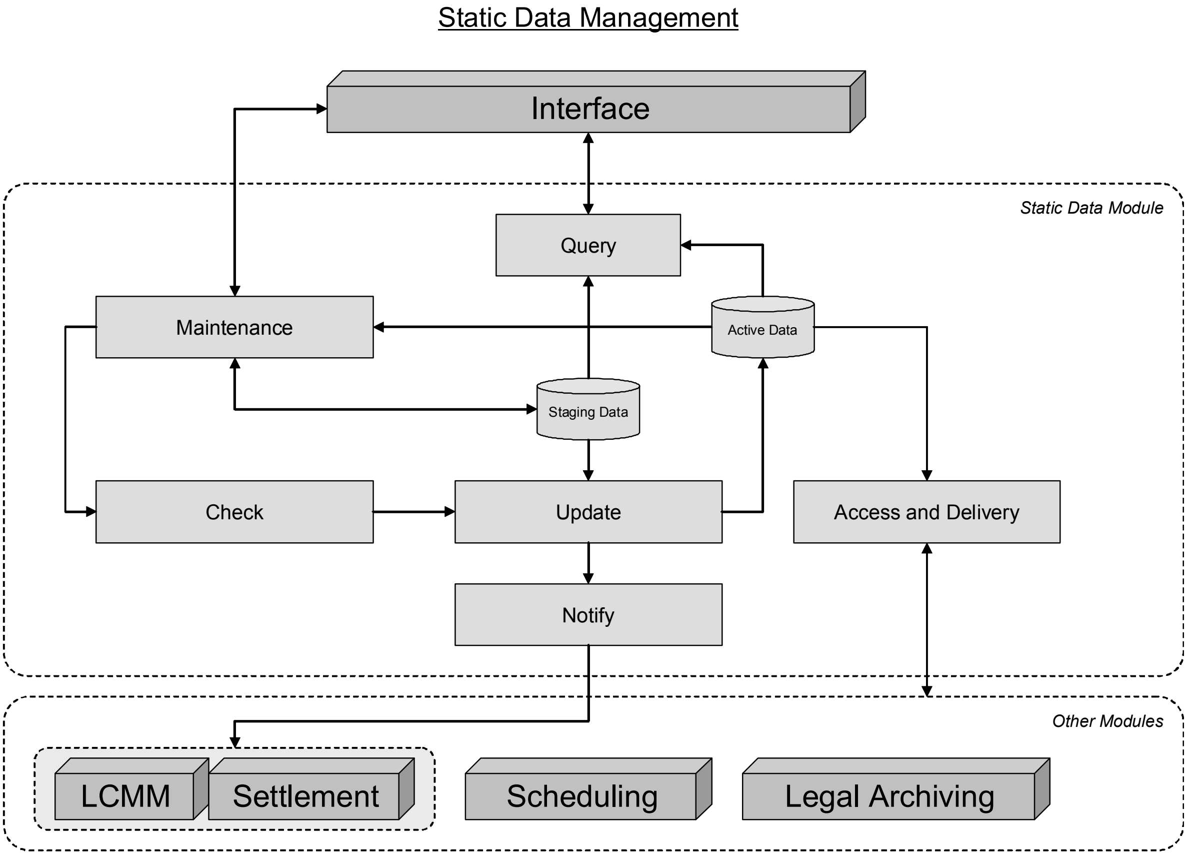


Figure 8 – Static Data Management architecture overview

The domain is responsible for the management of all the static data needed in T2S system. It allows external user, connected via the Interface domain, to collect and maintain Static Data information. The same Static Data domain enables the other T2S domains/modules to access to its data.

The "Maintenance" process is used to create, update and delete T2S static data. This process invokes the "Check" process to verify if a modification can be accepted or not. When no errors are encountered, the "Update" process is called in order to perform the maintenance of the database and, after that, if some notification is needed by other modules, the "Notify" process is invoked in order to produce the requested notice.

The "Check" process, depending on the kind of request, performs all the checks needed to ensure consistency of the database.

The "Access and Delivery" process provides the static data to the other domains/modules via dedicated interfaces. Each domain/module can have access to specific information in real-time, by using specific functions returning the current version of the requested data, or through data delivered on a periodic base (e.g. for legal archiving).

The "Query" process provides static data information to the users via the Interface module.

Synchronous or asynchronous interactions with the relevant modules (Query Management, Report Management, Statistical Information) are foreseen depending on the type of information requested by the users.

2.2.2.2. Processing principles

The Static Data Management must ensure the proper level of performances, minimising the risk to become a "bottleneck" for the T2S components which ask to access to its data. For this reason the domain is based on two basic processing principles:

- Updating transactions triggered by different system entities (e.g. a CB and a CSD or 2 different CSDs) are designed in a way to minimise the interaction between each other.
- Updating transactions are designed in a way to maximise the independency from data retrieval transactions.

These updates are carried out by the design processes shown in the figure above and by the design of the physical database, as briefly described hereafter.

All Static Data processes are implemented using COBOL language and uses, for transactional and messaging services, IMS/DC and WebSphere MQ. All accesses to the database are base on SQL language, both dynamic (to maximise maintainability) and static (when performances and stability have to be privileged). The DB2 stored procedures (written in COBOL language as well) are used to implement synchronous "read" access functions.

2.2.2.3. Data delivery

The domain makes available to the other domains/modules a set of synchronous interfaces (in pull mode) in order to provide detailed information on all the "active" data realising the "encapsulation" of the data itself.

The domain makes available to the other domains/modules a set of asynchronous interfaces[3] (in push mode) in order to provide the complete set of the static data. This allows the other domains/modules to update, or even completely re-create the possible mirror of the static data in the other domain's/module's databases where necessary (e.g. legal archiving). The used techniques for the interfaces are as described in chapter 2.1.2.3 "Internal technical Interfaces".

[3] Although this functionality is not explicitly mentioned into the GFS, the Static Data domain must provide an "export" functionality at least for the Archiving module (in order to allow the archiving of all Static Data). For this reason, the GTD foresees the possibility to put this Static Data "loadable set" at disposal of any other domains/modules.

2.2.2.4. Data maintenance and related internal technical interface

Maintenance principles

The domain makes available functions able to create/read/update/delete (CRUD) the detailed static data information.

The domain is able to manage only single requests for Static Data maintenance, when the management of massive set of requests (stored in a file) is needed (e.g. in the migration scenario) the Interface domain will split the set into single requests.

The CRUD functions can be invoked by the Interface domain in synchronous mode (in case of read access) or in asynchronous mode (in case of modification requests).

Notification

In some case a T2S domain/module need to be informed when some piece of information is modified. In these cases, on the basis of information set-up in a specific configuration table, Static Data sends a message to the "subscriber" domain module just after the update took place. The notified domain module has all the information (both on the notification message or invoking a specific "query" service) about the Static Data change that happened..

2.2.2.5. Data

Data objects

The domain owns its private[4] database to store and manage static data needed by all T2S domains/modules. This database stores all T2S entities related to:

- Parties
- Securities
- Securities Accounts
- T2S Dedicated Cash Accounts
- Rules and Parameters.

Data access and storage

Data are segregated in two types of physical tables.

Each Static Data table is split into an "active" table and a "staging" table, both with the same schema.

[4] I.e. only Static Data domain can read or update the Static Data database. The other T2S domains/modules can only ask services to Static Data Management domain.

The ”active” tables contains the confirmed versions with information available for the other domain/modules.

The ”staging” tables contains the not yet confirmed versions. They stores data that cannot be considered valid by the other domains/modules (e.g. for lack of confirmation in the 4-eyes flow or being validated for multiple cross-checks). The “staging” tables also store “old” revisions of Static Data instances.

The processes that access and update the ”staging” database are designed to avoid interaction with the functions which provide detailed information to the other modules/domains.

2.2.3. Lifecycle Management and Matching

2.2.3.1. Architecture overview

The Lifecycle Management and Matching (LCMM) domain deals with the lifecycle of instructions received in T2S. It consists of four core modules: Instruction Validation, Instruction Matching, Instruction Maintenance and Status Management:

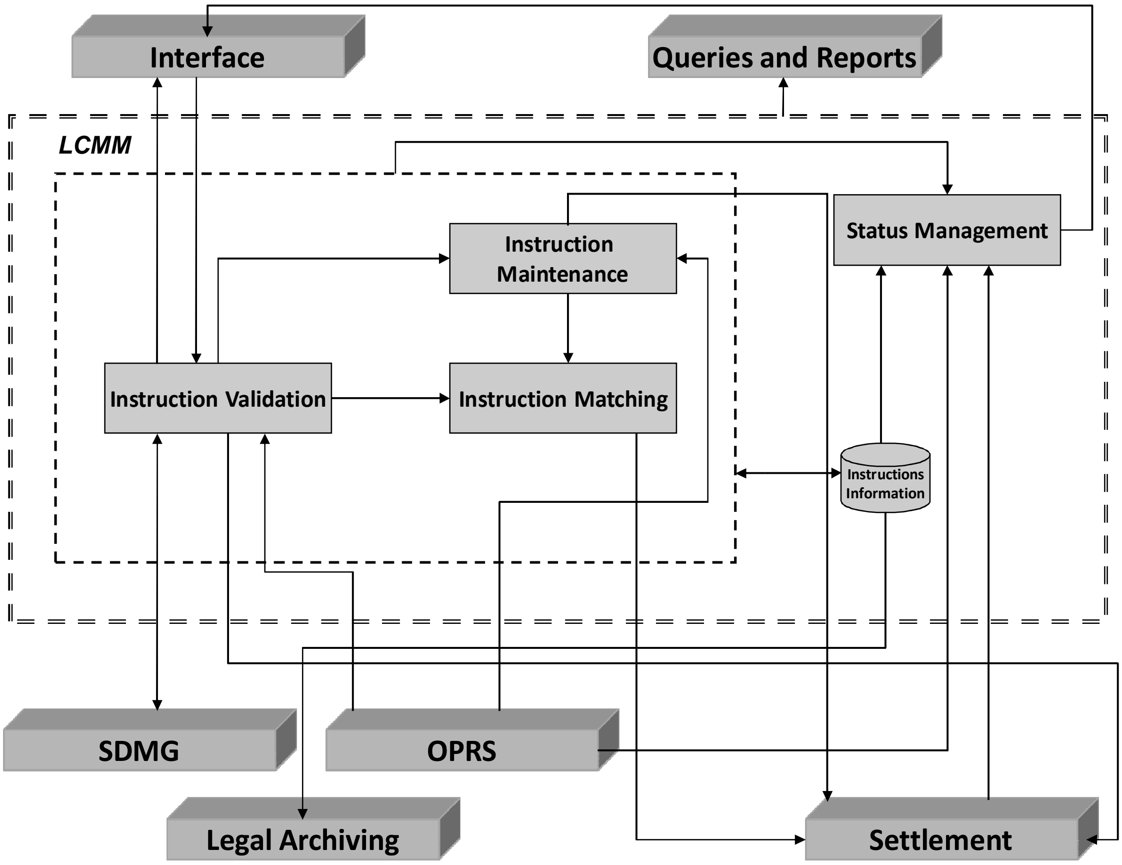


Figure 9 – Lifecycle Management and Matching architecture overview

2.2.3.2. Processing principles

LCMM processes run in three executing environments in line with the architectural design principles:

- IMS online regions to process single instructions in a continuous way
  - Instructions processing (validation, matching, maintenance) during the whole day, both for U2A and A2A interactions.

- IMS batch jobs to process large groups of instructions in a sequential and planned way, such as:
  - Selection of instructions to be revalidated at SoD (Start of Day) and due to Static Data changes.
  - Automatic cancellation of instructions at EoD (End of Day) which recycling period has expired.
  - EoD reporting.
- Stored Procedures at the database level to provide or obtain data in a synchronous mode between LCMM and other domains.

The asynchronous communication between the LCMM modules and between LCMM and other domains is done through IMS queues except with Interface, which is performed via MQ series, both for inbound as well as for outbound communication.

For synchronous communication between LCMM and other domains, as it is the case to obtain the most current information from Static Data, stored procedures are used. Within LCMM modules the standard Cobol "call" is used.

For the processing within the LCMM domain, the following principles are taken into consideration:

- **Parallelism**: In LCMM all processes (validation, matching and maintenance of instructions) run in parallel. Parallelism is a common behaviour with other T2S domains, which takes advantage of the Sysplex Technology. This design is highly scalable both horizontally and vertically to cope with future demands. In order to prioritise one process acting on an instruction over one another, certain mechanisms should be applied (i.e. check time-stamp or status of the instruction to determine which action should be performed first o not performed at all).
- **Concurrency**: In LCMM, two different processes might try to get hold of the same instruction at the same time. Concurrency is a consequence of different processes acting on the same piece of data (i.e. an instruction being revalidated is impacted by a maintenance action or the same instruction is being affected by several SD changes). For these situations, certain mechanisms like the usual DB2 locking facilities will be used.
- **Retry:** In some occasions, certain problems stemming from concurrency between different processes inhibit the processing of a message, for instance due to locks that result in deadlocks or time-outs. A case of time-out might happen when two different processes (i.e. maintenance and revalidation of an instruction) are trying to update an instruction at the same time. In these cases, one of the concurrent processes will be cancelled, rolled-back and re-tried later. IMS provides a facility to re-queue the input message of the cancelled process in the input queue.

- **Optimization of critical tasks**: Inside LCMM, critical tasks paths are prioritised, diverting secondary tasks to other processes which then execute asynchronously. For instance, preparation of validation messages addressed to end users is handled separately from the validation process itself, which is more critical than the notification to the user, so that validated instructions can be submitted as soon as possible to the Settlement domain for their settlement. In this case, the update of the instruction status, which triggers the next process in the live cycle of an instruction, is handled synchronously whereas the collection of instruction status information for the user is performed asynchronously.
- **Timeframe issues**: LCMM has to react in a different way depending on the phase of the business day (SoD, daytime period, maintenance window, etc.). Therefore it should be established what processes should take place or should be restricted in the transition from one phase to the other. For instance, at the beginning of EoD it should be well defined the sequencing of LCMM process in order to determine when the last instruction eligible to be processed during the current business day was successfully processed and thus set the real EoD event for LCMM.

2.2.3.3. Instruction Validation

Instructions entering the LCMM domain are validated before they are routed to the next modules for further processing. This validation is implemented as an MPP (Message Processing Program) process. Batch processes are only used to select existing instructions affected by Static Data changes and at SoD to be revalidated.

On-line Sample Scenario: Validation of incoming instructions

- Description: An instruction coming from the Interface domain has to be validated.
- Details: Processing is started by the presence of a message in a queue and it is executed as an IMS transaction. Multiple messages related to incoming instructions are processed in parallel. The results of the validation process are stored in the database and relevant information is submitted to different queues to be routed to the appropriate modules in the processing chain. These actions take place within the same u.o.w. (unit of work). As mentioned previously, these results are communicated asynchronously to the Status Management module to collect the information to be sent to the T2S actor.

Batch Sample Scenario: Selection of instructions to be revalidated at the Start of Day.

- Description: At SoD, instructions that that are not settled and have not been cancelled have to be selected to be revalidated.
- Details: The process is started by the Scheduler and it is executed as a BMP application, which will select the instructions to be revalidated and insert them into the input queue for the validation process. To optimize the process, a number of messages can be sent to

this queue in the same u.o.w. and, if necessary, the pace of inserting those messages could be controlled to avoid overloading the queues.

2.2.3.4. Instruction Matching

The Matching module compares the settlement details provided by the buyer and the seller of securities.

On-line Sample Scenario: Matching of a validated individual settlement instruction

- Description: Whenever an instruction is sent to the matching module, it is compared to all remaining unmatched instructions.
- Details: Processing is started by the presence of a message in a queue and it is executed as a MPP application. Optimized matching algorithms are considered in order to find the counterpart instruction in the repository of unmatched instructions as efficiently as possible.

2.2.3.5. Instruction Maintenance

The processes in this module are implemented entirely in online mode except for the function "cancellation by the system" which runs in a batch mode.

On-line sample scenario: Amendment of individual settlement instructions

- Description: Maintenance instructions intend to cancel, hold or release existing individual settlement instructions, amend business data of these instructions or even cancel cancellation instructions.
- Details: The process is triggered by the presence of a message in a queue and it is executed as an MPP application. Data changes are updated in the local database and relevant information is submitted to different queues to be routed to the appropriate modules. These actions take place in the same u.o.w..

Batch sample scenario: Cancellation by the system

- Description: Cancelling instructions which have failed to be matched or settled after a predefined period of time.
- Details: The process is triggered by the Scheduler (EOD Recycling and Purging time event) and executed as a BMP application.

2.2.3.6. Status management

This module is in charge of collecting the relevant data regarding status changes coming from the Settlement and LCMM domains and reporting them to the Interface domain. It processes the data related to these updates, collecting, if needed, additional information in order to forward them in a consistent order to the Interface domain.

On-line sample scenario: Collect data to inform of the validation status of an individual settlement instruction

- Description: Collects data to compose a status message informing about an update in the validation status of an individual settlement instruction and forwards it to the Flow Management module.
- Details: The process is triggered by the reception of a message in a queue and it is executed as a MPP application. Once the change in the instruction status is performed in the corresponding module, it is communicated asynchronously to the Status Management module, along with additional information to compose the status advice to the user.

2.2.3.7. Data

Data objects

LCMM manages instructions-related data such as:

- LCMM Instructions (Settlement Instructions, Settlement Restrictions, Maintenance Instructions)
- Instruction Status History
- Reason Codes History
- Unmatched Settlement Instructions
- Instruction Links
- Allegements

It also makes use of static data, which is accessed through stored procedures.

Data access and storage

Data managed by the LCMM modules is stored in a DB2 database and is accessed by the processes running within the module via native SQL calls or stored procedures. LCMM data required by other domains is accessed via stored procedures.

2.2.3.8. Internal technical interfaces

The interfaces used in the LCMM domain can be summarized as follows:

- Communication in synchronous mode between Cobol programs (program-to-program call) within the LCMM domain.

- IMS queues for an asynchronous communication between LCMM Modules (i.e. the Validation module sends a successful validated Instruction to be matched in the Matching module) and between LCMM and other domains (i.e. a matching object is sent to Settlement).
- MQ queues for asynchronous communication between LCMM and the Interface domain.
- Direct synchronous access to LCMM data base.
- Stored Procedures for synchronous access to data managed by other domains or to provide other domains with LCMM data.
- Flat files with information extracted from the LCMM data base for the Legal Archiving domain.

2.2.4. Settlement

2.2.4.1. Architecture overview

The Settlement domain deals with the settlement of settlement instructions and matching object (pair of matched settlement instructions) received from Lifecycle Management and Matching (LCMM) and Liquidity Information issued from Liquidity Management. The Settlement domain could also deal with settlement instruction maintenance requests or settlement restrictions received from LCMM. It consists mainly of five modules, Standardisation and Preparation to Settlement, Daytime Validation Provisioning and Booking, Auto-collateralisation, Daytime Recycling & Optimisation and Night-time Settlement.

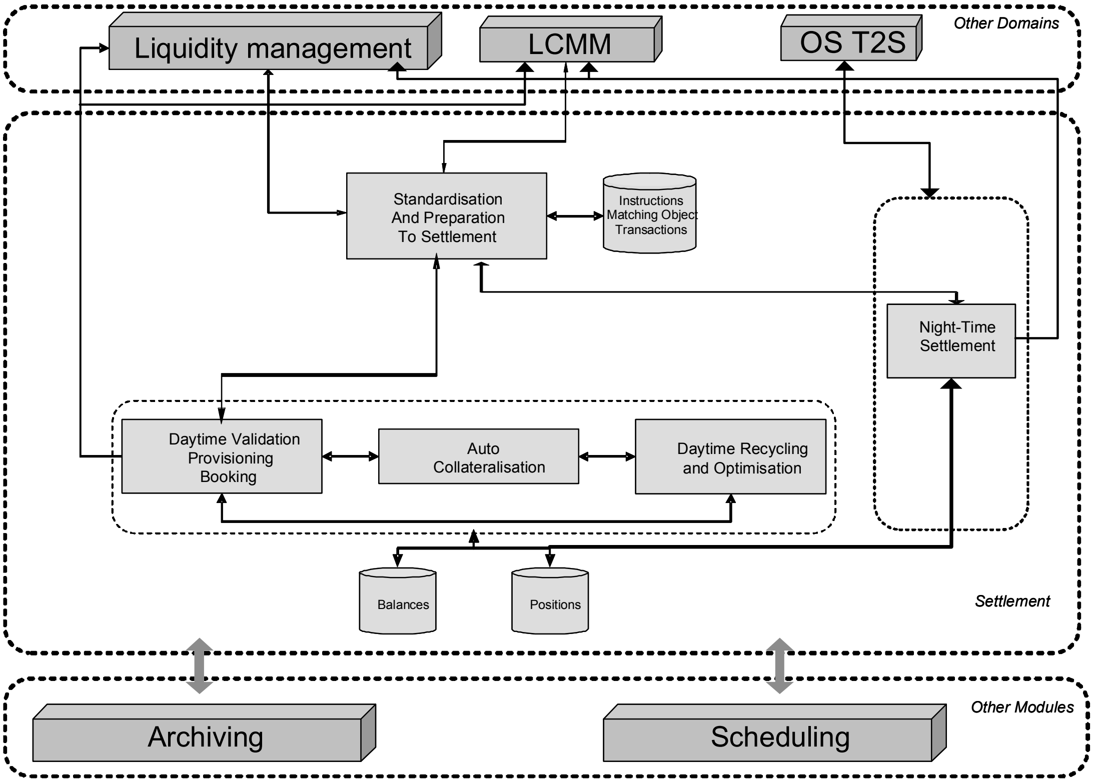

Liquidity management
LCMM
OS T2S
Other Domains
Standardisation And Preparation To Settlement
Instructions Matching Object Transactions
Night-Time Settlement
Daytime Validation Provisioning Booking
Auto Collateralisation
Daytime Recycling and Optimisation
Balances
Positions
Settlement
Archiving
Scheduling
Other Modules

Figure 10 – Settlement architecture overview

2.2.4.2. Processing principles

Standardisation and preparation to settlement

Introduction

2.2.4.3. The main aim of this module is to standardise the instruction and theirs respective matching objects into elementary transactions which are better suitable for a smooth processing by the settlement engine thus allowing the system to fulfil the performance requirements using notably message driven processes (MPP) relying on IMS scalability features. (see § 2.1.2.4 General architecture – back-end section related to IMS parallel processing capability and horizontal scalability). This module also deals with maintenance requests and settlement restrictions.

Processing

This module receives settlement instructions and matching objects from Instruction Validation and Instruction Matching modules and constitutes settlement transactions and settlement collections of transactions to be sent to the Daytime Validation and Provisioning Booking module during the day and to the Night-Time Settlement module during the night. When it receives settlement restrictions or instruction maintenance requests from LCCM modules, it constitutes dedicated and specifics settlement transactions and settlement collections of transactions. Executed as MPP application (day and night), it sends prepared settlement transactions and settlement collections of transaction in an asynchronous way to the Daytime Validation and Provisioning Booking module (in day) or to the Nigh-Time Settlement module (in night). Furthermore, this module should generate by himself settlement objects and settlement instructions (i.e. for realignment chain process). The particularity is that these entities have to been shared with LCMM domain.

Daytime Validation, provisioning and booking

Introduction

This module is running only during day time period. It is the core module of the "Settlement domain" and of the system. It must be able to process a high volume of data in a high performing way and a short period of time because the overall efficiency of the system depends on it.

For these reasons, the functions implemented have to take advantage of massive parallel processing features (see § 2.1.2.4 General architecture – back-end section related to IMS parallel processing capability and horizontal scalability). The aim is to support the volume of data announced and to have a positive impact (parallel processing e.g starting a VPB as soon as the technical resources are available avoiding a kind of sequential (bottleneck) processing) on the duration of the overall settlement process for a new instruction coming into the system. Parallel processing (especially use of shared resources such as cash accounts for instance) implies an in-depth analysis especially regarding the use of required data because of possible locks implied by such a strategy (this point is discussed in the following chapters – see § 2.2.4.7 Parallel and concurrent processing – notably deadlock avoidance).

To take advance of the parallel processing capabilities of the system (2.1.2.4 General architecture), the major part of the processing is implemented using IMS MPP transactions). These MPP transactions are designed and implemented in a way to minimise the duration of their own execution time with optimised coding.

Processing

The critical step of the overall process is the booking along with the update of the cash balances and the security positions implemented as an MPP transaction. This is done in a way that deals in a safe way with concurrent accesses (updates) on the same objects and that absolutely avoids deadlocks (refer to 2.2.4.7 Parallel and concurrent processing – DB2 and application design), timeouts and long lasting processes.

At the end, the feedback to other modules (within and outside the Settlement domain) of the result of the booking process is managed (status etc) via a IMS Message Queue, taking care notably of the overall consistency of information addressed out of the application.

2.2.4.4. Daytime Recycling and optimisation

Introduction

This module is running only during day time period. It has the objective to improve the settlement rate of the overall system. It is able to detect combinations of transactions which can be settled if considered together. This is useful for detection of certain market situations (back-to-back transactions, circles…) when trying to settle each transaction separately, whatever the original order of submission to the booking would lead to fails.

The module is triggered by business events (settlement success or failure). He process recycling on settlement success event trying to build a settlable solution within and with remaining unsettled transactions. He process settlement optimisation on settlement failure event trying to build a settlable solution within and with remaining unsettled transactions.

Functionalities of this module have to be implemented in a "transactional" way (e.g as a BMP/MPP under the control of IMS and in concurrency with other IMS transactions for instance VBP running in parallel etc). as regards optimisation algorithms launched on a business event. Parallelism between algorithms executions has to be taken into consideration in the overall design of this module.

Processing

Once the collection of all transactions belonging to the same sequence is identified, the Daytime Recycling and Optimisation (R&O) module starts the required optimisation algorithms.

The execution of an algorithm has to take into account specifics internal parameters and external parameters. These parameters have to be passed thought to the algorithms at launching and checked by himself during its execution (i.e. max duration of it execution).

2.2.4.5. Auto-collateralisation

Introduction

This module is running only during day time period. It has to provide the auto-collateralisation facility which permits a participant who has a need for cash in order to settle one or several transactions, to obtain an intraday credit from his National Central Bank.

This module has a different behaviour based on the way it is triggered: in case it is called by VPB module, it has to create settlement transactions that put in place the intraday credit and the collateralisation of a part of participant's securities whereas when called by Daytime optimisation algorithms, it just has to respond if the auto-collateralisation facility permits the settlement of the current set proposed by the algorithm (process in simulated mode).

Functionalities have to be implemented in a "transactional" way (e.g as a BMP/MPP under the control of IMS and in concurrency with other IMS transactions). These functionalities" have to be called in case of need (inserting a message in a IMS message Queue) by VPB and R&O modules. In case of forced reimbursement, the same calling method is used in direction of Liquidity Management.

Processing

Apart from the generation of the transactions representing the collateralisation and the credit themselves, a large part of the functionalities are independent of the triggering mechanism.

2.2.4.6. Night-time settlement

Introduction

This module is running only during night time period. It is a mix between the Daytime Recycling & Optimisation module (without recycling part) and Daytime Validation & Provisioning Booking module. It aims to optimize and settle during the night, remaining daytime unsettled transactions and all new settlement transactions received. Its execution has to be organized in successive sequences, each sequence cover a settlement transactions type perimeter. So it is envisaged to implement it in a "batch" way (as IMS BMP – Batch processing multiple messages). However, different from Daytime Recycling & Optimisation module, in addition it has to perform other more complex algorithms (i.e. Deselected type), in a big more high volume of received data context then day time. So some parallel processing solutions have to be analyzed.

Processing

The Night-Time Settlement module meets the same technical problematic already rose for the three last settlement modules described; Daytime Validation provisioning and Booking, Daytime Recycling and Optimisation, Auto-collateralisation modules. So it should resume their implementations.

2.2.4.7. Parallel and concurrent processing

According to the need of a settlement system which should be able to settle transactions without interruption and in a high performance way, the settlement engine is based on an IMS transaction manager which allows parallel processing across several IMS regions (2.1.2.4 General architecture).

Due to parallel processing, the technical data organization (mainly based on a DB2 storage), must be designed in order to avoid conflicts and long waiting periods in concurrent context. In fact, waiting periods cannot be avoided when two processes intend to access in exclusive mode the same data, but can be managed with respect to the requirements expressed.

As a general principle, concurrency management will rely on relational database capabilities in this area.

IMS based application

The settlement domain will be implemented according to the principles exposed in chapter 2.1.2.4 – back end using as well single message processing IMS MPP for fast processing such as VPB), than Batch messages processes IMS BMP) notably for optimisation processes and SOD/EOD treatments.

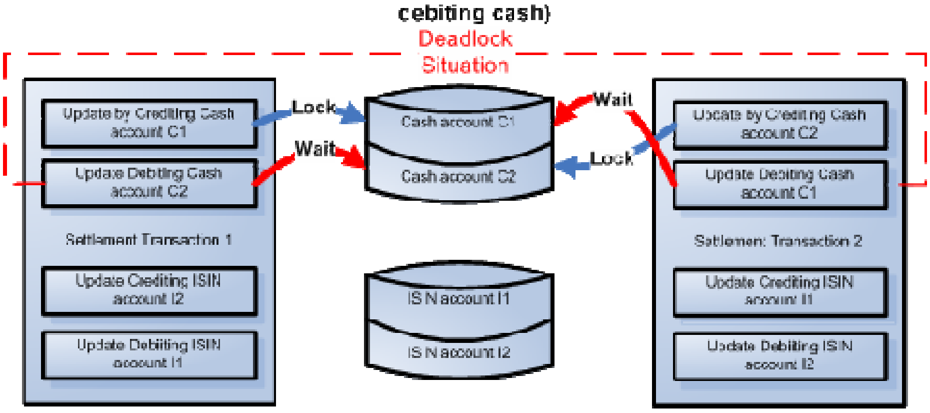
DB2 and application design

Because of parallelism and unavoidable lock constraints on objects, deadlock situations have to be managed in consequence.

A deadlock situation occurs when two processes try to lock two or more same objects in a different order (case 1), causing a situation without solution. This could happen if the order of locks is not managed at the level of application and relies only on technical features.

A way to prevent from this situation is to sort the objects to lock using an ordering key (identifier). It will cause only waiting state until the first process has finished (case 2).

Order of treatment dependant only of type of action (crediting cash, debiting cash)


Deadlock
Situation
Update by Crediting Cash account C1
Update Debiting Cash account C2
Settlement Transaction 1
Update Crediting ISIN account I2
Update Debiting ISIN account I1
Lock
Wait
Cash account C1
Cash account C2
ISIN account I1
ISIN account I2
Wait
Lock
Update by Crediting Cash account C2
Update Debiting Cash account C1
Settlement Transaction 2
Update Crediting ISIN account I1
Update Debiting ISIN account I2

Order of treatment dependant on Id of account:
The Transaction 2 will be treated when the first one will be finished

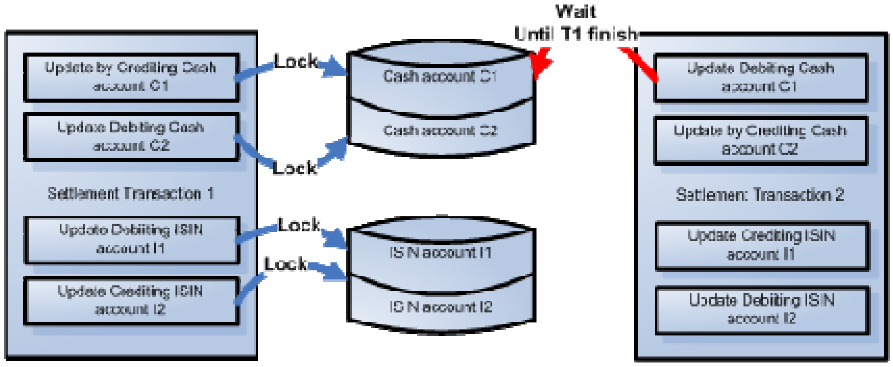

Wait
Until T1 finish
Update by Crediting Cash account C1
Update Debiting Cash account C2
Settlement Transaction 1
Update Debiting ISIN account I1
Update Crediting ISIN account I2
Lock
Lock
Lock
Lock
Cash account C1
Cash account C2
ISIN account I1
ISIN account I2
Update Debiting Cash account C1
Update by Crediting Cash account C2
Settlement Transaction 2
Update Crediting ISIN account I1
Update Debiting ISIN account I2

Figure 11 – Example: avoidance of deadlock situations

2.2.4.8. Data

Data objects

The following data is managed directly within the settlement domain:

- Settlement instructions;
- Matching objects;
- Settlement transactions;
- Settlement collections of transactions;
- Cash Balances;
- Security positions.

Other data:

- Data mandatory for processing such as static data.

Data access and storage

Regarding data storage, a relational database is mandatory (as defined as a general principle for the entire system) and some parts of the database have to be designed with regards to specific data of the Settlement domain (for example instructions, matching objects or transactions).

2.2.4.9. Internal technical interfaces

Most of the time, chains of IMS transaction calls are used. The communication between modules within the Settlement domain relies on the IMS transaction manager in order to decouple the processing and to benefit from the IMS parallelism. Concerning the data shared between the modules, usage of the database is envisaged, in order to guarantee the integrity of the data and a fast recovery in case of problems.

2.2.5. Liquidity Management

2.2.5.1. Architecture overview

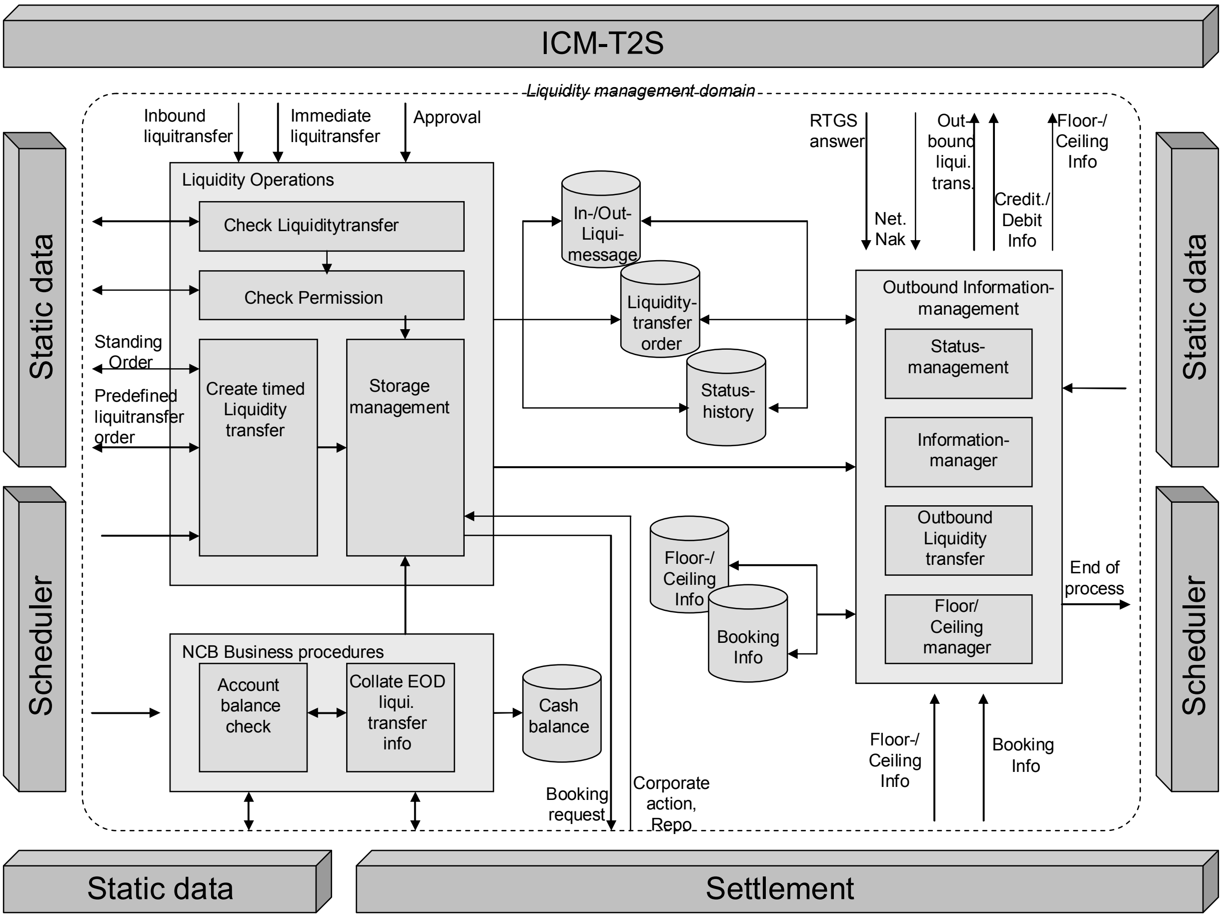


Figure 12 – Liquidity Management architecture overview

The "Liquidity management domain" is the overall instance responsible for all allocation and withdrawal of liquidity within T2S. It manages the necessary flow between external and internal liquidity resources. External liquidity resources are RTGS-systems from where the necessary liquidity required to foster a smooth processing of T2S is provided and where the surplus liquidity is sent to during or at the end of day. The component has interfaces to the "ICM-T2S domain" connecting it to the RTGS-systems and to T2S-participants, to the "Settlement domain", to the "Static data domain" and to the "Scheduler module" of the "Operational services domain". The overall functionality of the domain is logically divided into modules which are "Liquidity Operations", "Outbound Information Management" and "NCB business procedures".

2.2.5.2. Processing principles

The domain is decoupled from other domains by using asynchronous interfaces. Internal communication is done as described in chapter 2.1.2.3 "Internal technical Interfaces".

Roughly the workflow within the domain can be described as follows:

The data of transfer orders which enter the domain via the "ICM-T2S domain" are sent by messaging middleware which also triggers the adequate function of the "Liquidity management domain". This function stores the data flow in a DB2-table and calls subordinate functions by IMS-message-switch. The control of the order processing is done by internal status changes. Regarding standing- and predefined orders and the liquidity retransfer at EOD the triggering is done by the scheduling module. The technical implementation of the triggering is done by IMS or stored procedure. In this case the necessary processing data comes from the "Static data domain".

Liquidity Operations

The module comprises functions that are responsible for the validation of the liquidity transfer orders, for the management of the workflow of these orders and for the creation of booking requests to be sent to the "Settlement domain".

The data of liquidity transfer orders that have to be validated arrive from the "ICM-T2S domain" as liquidity transfers from RTGS systems or U2A- or A2A-transfer orders from T2S customers .

The module is also responsible for the creation of liquidity transfer orders which are initiated by scheduler in case of time or business events during the day or at EOD processing.

In this case the control data needed for the processing is stored and validated by the "Static data domain". During the start of day procedure the "Static data domain" provides all the necessary data (account data, participant data, predefined and standing orders, etc.) to the "Liquidity management domain" via the internal technical interfaces which are realized as defined in chapter 2.1.2.3 "Internal technical Interfaces".

Here is an exemplary flow of the domain:

- a RTGS system sends a liquidity transfer order message via "ICM-T2S" to the "Liquidity management domain". After authentication checks ICM-T2S writes the transfer order into a dedicated MQ queue and triggers the "Liquidity operations module"
- the module stores the message and creates an order with an initial processing status
- then the order is validated and permission checks according to the user role concept are performed
- in case that all checks are passed successfully the next function creates an booking request and sends it via IMS message to the "Sequencing and prioritization module" of the "Settlement Domain"

- the "Validation, Provisioning and Booking" module of the "Settlement" domain returns the result of the booking request via IMS message to the "Outbound information management module" of the "Liquidity management domain"
- this module updates the status history of the transfer order and notifies the sending RTGS system about the booking. Therefore a information message is created and sent via MQ to the "Outbound processing module" of the "ICM-T2S domain" which forwards it to the RTGS system
- additionally, the "Outbound information manager" sends a booking information to the concerned account holders via the same technical way if requested.

Outbound Information Management

The task of this module is to maintain the status of liquidity transfer orders and to create the outbound liquidity transfer message to connected RTGS systems. Moreover it provides booking information, floor/ceiling information and information about RTGS rejections to the T2S actors.

The module has interfaces to the internal "Liquidity operations module" and externally to the "Settlement domain", the "ICM-T2S domain", "Static data domain" and the "Scheduler module".

Except towards the "ICM-T2S domain" where Websphere MQ as interface technology is used in all other cases IMS program to program switch is applied for communication.

An exemplary flow of the module looks like:

- the module is informed by the "Settlement domain" about the successful settlement of a immediate liquidity transfer bound for a RTGS system
- at first the status of the liquidity transfer order is updated to "booked"
- then all information necessary to create an outbound liquidity transfer messages for the concerned RTGS are assembled from the "Static data domain"
- the created outbound liquidity transfer message is sent to the "ICM-T2S domain" via Websphere MQ to be forwarded to the respective RTGS system
- further on the module checks in "Static data" whether the T2S actor of the debited account has opted for booking information
- If that's the case all the information necessary for creating a booking information message is assembled from "Static data" and a information message is sent to "ICM-T2S" via Websphere MQ bound for the T2S actor.

NCB business procedures

The module is responsible to balance the T2S liquidity accounts to zero at EOD. It is triggered by the "Scheduler module" of the "Operational services domain". It interfaces the "Liquidity Operations module" of its own domain, the "Settlement domain" and the "Static data domain". According to static data the module requests the balances of all cash and transfer accounts from the "Settlement domain". Then it calls the internal "Liquidity operations module" to perform the following processing in the given order:

- if there isn't enough liquidity on a cash account to reimburse the total amount of intraday credit, a liquidity withdrawal from the respective RTGS system is initiated
- the transfer of surplus liquidity on cash accounts towards the connected RTGS systems is initiated
- the clearing of the transfer accounts towards a technical RTGS account in T2S is initiated if the balance is not zero

All these EOD procedures are supplied on a currency basis. This means that if closing times of a business day differ from currency to currency, the NCB business procedures will be scheduled at different times.

An exemplarily flow of the domain looks like:

- at EOD the scheduler module invokes a function of the "NCB business procedures module"
- the function addresses to "Static data" and requests the balance of all liquidity accounts from the "Settlement domain"
- the "Settlement domain" returns the balances and the "NCB business procedures module" forwards them to the internal "Liquidity operations module"
- there the liquidity transfer orders are created and the booking requests are sent to the "Settlement"
- the "Outbound information management module" receives the answer of the booking request from the "Settlement domain" updates the status of the order, creates an outbound liquidity transfer and sends it via the "ICM-T2S domain" to the related RTGS system
- Moreover if requested it sends a booking information to the concernedT2S actor

2.2.5.3. Data objects

The necessary data is stored in the tables of a local database owned by the domain. The database contains tables containing:

- In- and outgoing messages (from/to other domains)
- booking orders (to settlement)
- protocol table
- static data (replicated from static data domain)
- account balances

Data access and storage

The data access to the tables is done by SQL and a separate access layer is envisaged. Requests to data of other domains are done via IMS, Websphere MQ or stored procedures. This can happen in pull or push mode.

2.2.5.4. Internal technical interfaces

The internal communication is realized as defined in chapter 2.1.2.3 "Internal technical Interfaces". The necessary processing data is kept in database tables.

2.2.6. Statistical Information

Two implementations of this module are considered, respectively hosted:

- in region 1/2 associated with the Short Term repository;
- in region 3 associated with the Long Term[5] repository.

Both implementations will be based on the conceptual architecture described herein[6].

2.2.6.1. Architecture overview

The following diagram represents the software objects implementing the Statistical Information module functionalities, the database and the main internal and external data flows. Of course the software objects are identified starting from the functionalities, nevertheless there is not a 1 to 1 relationship between functions and software objects.

[5] Short Term and Long Term repositories are described in Chapter 2 section 2.2.6.1 of the T2S General Specifications.

[6] Note that this architecture has not impact on the Query and Reports modules (that have different goals and/or scope). There is only one Query Report and only one Report module; they are hosted in region 1/2

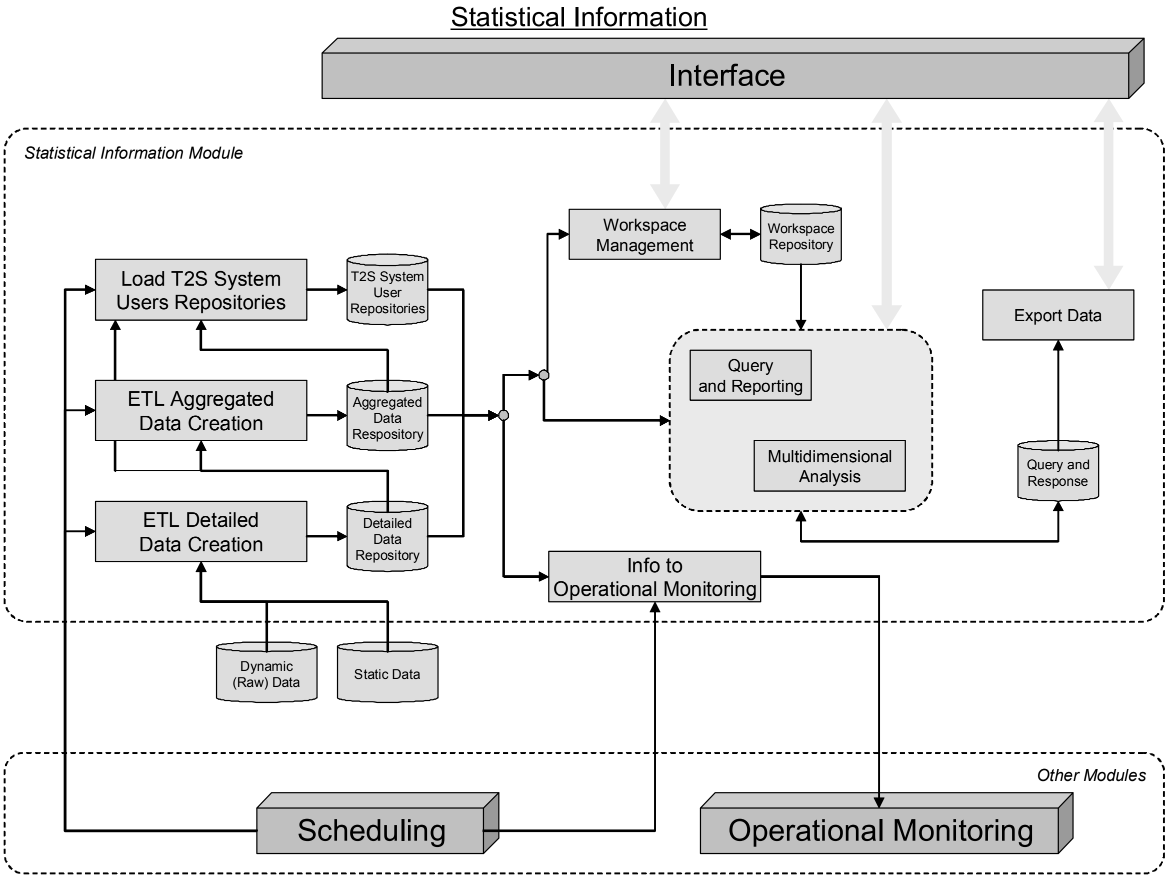

Statistical Information
Interface
Statistical Information Module
Workspace Management
Workspace Repository
Load T2S System Users Repositories
T2S System User Repositories
ETL Aggregated Data Creation
Aggregated Data Respository
ETL Detailed Data Creation
Detailed Data Repository
Query and Reporting
Multidimensional Analysis
Export Data
Query and Response
Info to Operational Monitoring
Dynamic (Raw) Data
Static Data
Other Modules
Scheduling
Operational Monitoring

Figure 13 – Statistical Information architecture overview

Short-Term Data Repository is updated on a regular basis to reflect on-line data up to three months. The Long-Term Data Repository and the other databases are updated by periodic processes (triggered by the Scheduling). The module's database data is a copy of the other domains/modules data. There are 3 databases with different levels of aggregation.

The Workspace Management process allows the user to define interactively the business views of the data. The information is stored in the Workspace Repository.

The Query and Reporting and Multidimensional Analysis processes retrieve the information requested by the user (the info can be seen interactively or exported). The module provides also information to the Operational Monitoring module.

2.2.6.2. Processing principles

- The data is stored in a separate environment;
- the processes to load and transform the Detailed Data Repository, Aggregated Data Repository, Users Data Repository, Statistical Workspaces, are separate and autonomous from the processes to retrieve the data.

2.2.6.3. Data

Data objects

The module has more than one database receiving periodic (mainly daily) copies of the dynamic data of the other domains/modules :

- **Short Term and Long Term Data Repositories**, with the same granularity as the operative databases. The design of the physical database starts from the conceptual data model and takes into account also the performance needs linked with the use of the data in the Statistical Model. Also the design of the statistical data model is, in principle, independent from the operational data model.
- **Aggregated Data Repository**, with a lower level of granularity compared to the operative databases. The design of the physical database is realised on the base of the functional specification.
- **Users Data Repository**, data of interest of specific categories of T2S system users. The design of the physical database is realised on the base of the functional specification.
- **Statistical Workspaces**, with the business-oriented concepts and their mapping to the physical objects.

Data access and storage

The module owns its private databases. The data stored in the Detailed Data Repository and Aggregated Data Repository are structured in a way to make the multidimensional analysis easier.

2.2.6.4. Internal technical interfaces

The module has a set of technical interfaces with the Interface domain. These technical interfaces work in synchronous or asynchronous mode, depending on the type of request/function (Query, report, management, data extraction).

In principle, the technical interfaces with the Scheduling module and the Operational Monitoring module are designed to be asynchronous.

The Extraction, Transformation and Loading function is triggered at the End-of-Day to load on a daily basis the relevant data into the Short-Term and Long-Term repositories. The only exception is the first step of the ETL process that in real-time loads data in the short term data store.

2.2.7. Query Management

2.2.7.1. Architecture overview

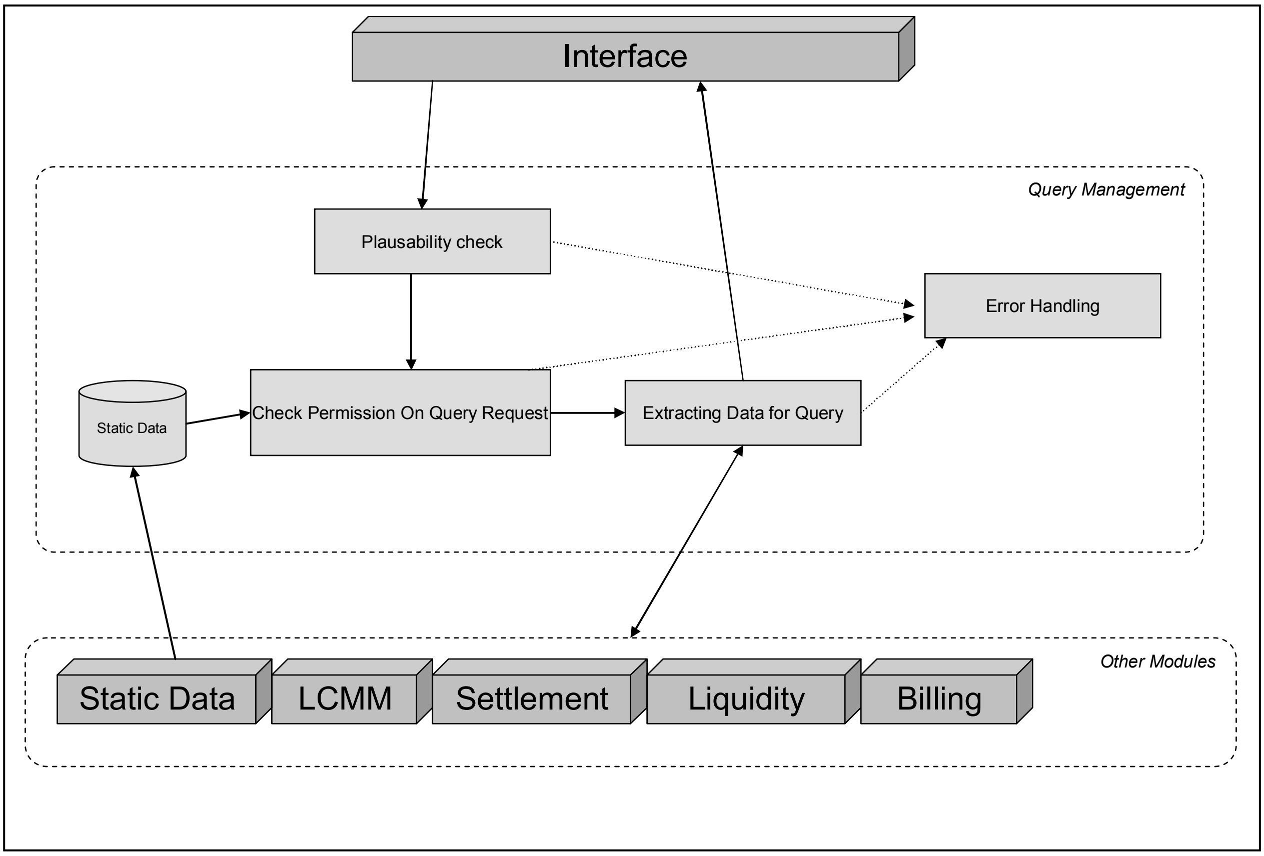


Figure 14 – Query Management architecture overview

The Query management is part of the SQRA (Statistics, Queries, Reports and Archive) domain. It provides the possibility to the participants to request real time and historical queries. The queries are pre-defined, nevertheless several search fields can be specified and combined via logical operators and wildcards.

The queries are requested via the Interface domain by the participants using U2A mode as well as A2A mode.

2.2.7.2. Processing principles

The communication with the external partners, CSDs and directly connected T2S parties is using the Interface domain. The data of the response is handed over to the Interface Domain where the conversion from and to XML in compliance with ISO 20022 is done using its standard features..

U2A and A2A Queries will be optimised, means that there will be own interfaces for each approach if there are differences in the input or output parameters of the queries, even if they are rather similar and if they are handling the same business scenario. E.g. for the U2A approach there could be two steps on the screens, the first requesting a list of items, only with basic information and the second requesting detailed information for one of these items. Whereas for the A2A approach all information will be joined together in one request. For that scenario three interfaces to the backend will be used. One for A2A, one for the U2A list and one for the U2A detailed information.

2.2.7.3. Workflow

The query coming from the Interface domain is first routed through a basic plausibility check, then the data based permission checks are done. After that the relevant domains are called in a synchronous mode collecting all information in order to guarantee a consistent state of the queries, especially for domain spanning queries. The interfaces are optimised to reach the best possible performance, e.g. there are different interfaces for similar queries coming from U2A and A2A. This response information is then sent in the response to the Interface domain.

2.2.7.4. Data

The result data of the query is not stored inside the Query Management Module.The data will only be routed through the Query Management to the Interface Domain where the XML generation is done.

2.2.7.5. Internal technical interfaces

The communication to the Java based Interface domain is done using EJB calls in a synchronous mode.

As all queries have to respond in real time to the user, all interfaces to the queried COBOL based domains are synchronous using Stored Procedures. Only Balance queries in A2A mode acting during a settlement cycle are queued, but that is already done inside the Interface Domain and resent after the cycle. Nevertheless also here Stored Procedures are used.

2.2.8. Report Management

2.2.8.1. Architecture overview

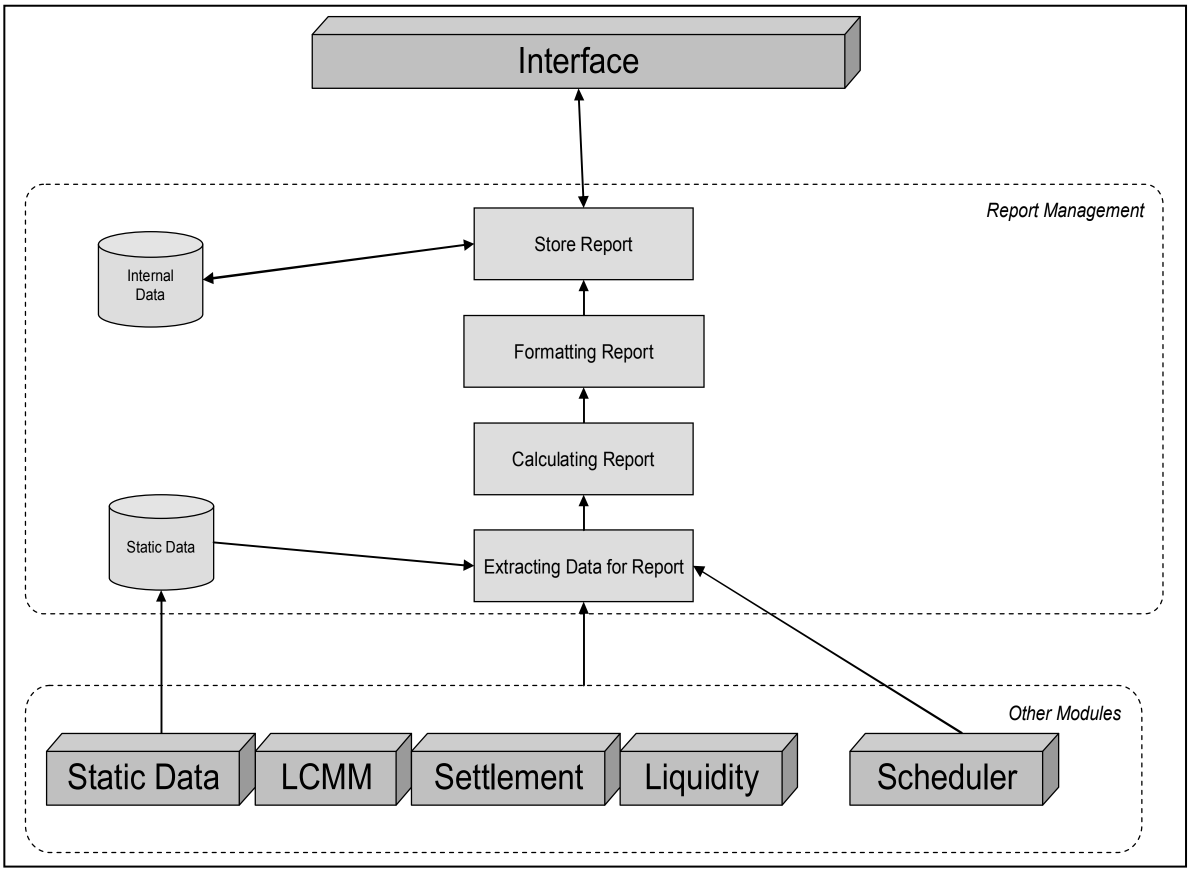


Figure 15 – Report Management architecture overview

The Report management is part of the SQRA (Statistics, Queries, Reports and Archive) domain. It provides predefined reports about production data to CSDs and directly connected T2S parties. Therefore it needs a direct communication to all domains owning the relevant data. The creation of all reports is initialised by events sent from the Scheduling module..

2.2.8.2. Processing principles

The Report Management Module starts the creation of a report, initialised by an event from the Scheduling module by calling the relevant Backend Modules in a synchronous mode. If more than one Backend Module is referenced and consistency of the data over these modules has to be guaranteed, the calls are done inside one single transaction. The consistency of data inside one Backend Module is guaranteed within itself.

For each report of the various T2S Actors the special requirements are stored as report configuration inside the Static Data Domain.

2.2.8.3. Workflow

After the event from the Scheduling domain the creation of a report series is started. For each report, the list of T2S Actors getting this kind of report is determined and for each of that single reports the requested parameters are find out. For both investigations the Static Data Domain is storing the needed information as report configuration and is therefore requested real time using the static data interfaces. Now for each single report the referenced Backend Modules are called for the information . As now the Report Management Module has collected all data for that report it does some additional calculations to complete the requested information, different for every kind of query based on the functional requirements. The next step is the formatting (e.g. sorting, grouping) of the data and the creation of the XML files, compliant with the ISO 20022 standard. These reports are stored on disc. Additionally for each report an entry into a database table is done where some additional information for this special report is saved, e.g the id, the name, date, the placement.

At last the reports are forwarded over the Interface Domain to the actors that have requested that in the report configuration. From now on these reports can be downloaded by the participants that have subscribed for that. Therefore the request is coming over the Interface Domain to the Report Module in order to encapsulate and harmonize the access. A direct access to the files is not supported.

2.2.8.4. Data

All generated reports are stored as XML files within the Report Management on disc.Meta information for the reports are stored in the database.

2.2.8.5. Internal technical interfaces

The communication to the Interface domain is done using EJB calls in synchronous mode.

For the communication between the Report Management and the domains owing relevant data synchronous calls using Stored Procedures are employed.

.

2.2.9. Billing

2.2.9.1. Architecture overview

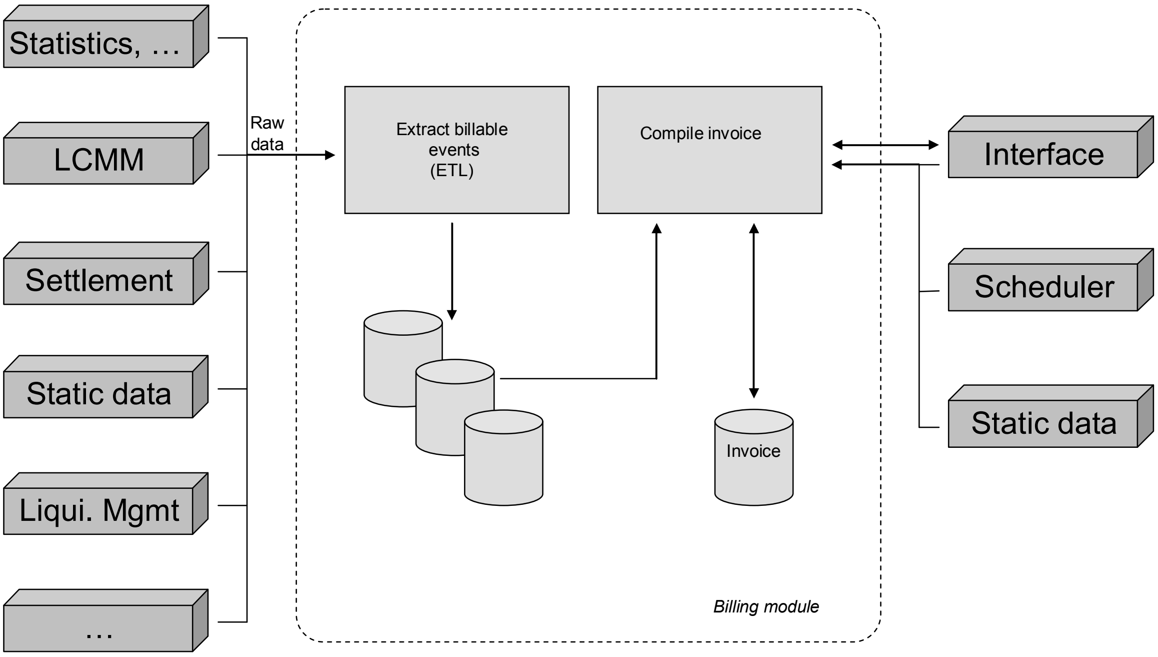


Figure 16 – Billing architecture overview

The billing module is part of the "Operational services domain". It has no internal interface to other modules of this domain. The functions are implemented as COBOL IMS programs (MPP and/ or BMP). Communication to and from other domains is done by Websphere MQ and by sequential files.

2.2.9.2. Processing principles

Every EOD the data of all potentially billable events of T2S are read by the billing module from files the domains create where they occur during the business day (this might just as well be the data bound for the "Statistical module"). Within the "Billing module" a function is triggered which takes the data (raw data) over into the local database of the module.

At the start of a new business month the function of the module is triggered by the "Scheduler module" and produces the invoices for all participants for the previous month. The invoices are stored in a local database table. Every participant can request an invoice for one of the last 12 months or for the current month until the last business day by U2A or A2A request.

2.2.9.3. Data

The raw data necessary for the processing of invoices is send by the domains where billable events occur. This data is stored in local tables of the module:

- the raw data of the billable events stemming from different domains;
- the generated invoices.

Data objects

The data objects that are used by the "Billing module" are the tables of billable events coming from "LCMM domain", "Settlement domain", "Static data domain", "Liquidity management domain", and "Statistics, reports, queries and archive domain".

Data access and storage

The necessary processing data is stored in local database tables and is accessed by SQL instruction out of the programs (functions). Data coming from other domains are processed via sequential files. The triggering of the transfer of the raw data is done via IMS or Websphere MQ.

2.2.9.4. Internal technical interfaces

The internal communication is based on IMS program to program switch. The necessary data for processing is retrieved from local database tables.

2.2.9.5. Annotation

If more precise functional specifications show that the necessary data for the generation of invoices can be received from the "Statistical information module" the step of gathering raw data will be dropped. In this case the information of billable events will be directly requested from the "Statistical information module".

2.2.10. Legal Archiving

Architecture overview

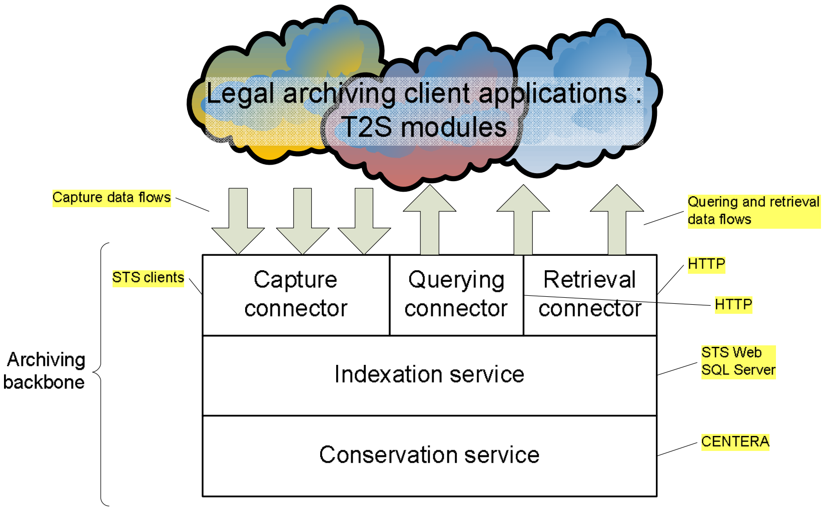


Figure 17 – Legal Archiving architecture overview

The solution consists of the following components/services:

- Connector for capture: standardises the method of access to the infrastructure. Documents to archive are sent by every module to a dedicated storage area;
- Connector for consultation: allowing the presentation of lists of documents corresponding to the research criteria;
- Connector for restoration / export: with the aim of the extraction of one or several archived documents;
- The indexation service allows the correspondence between the document and its localisation within the technical infrastructure of storage of archives;
- The conservation service offers a value-added storage, being compliant with the standards and the rules of legal archiving.

2.2.10.1. Processing principles

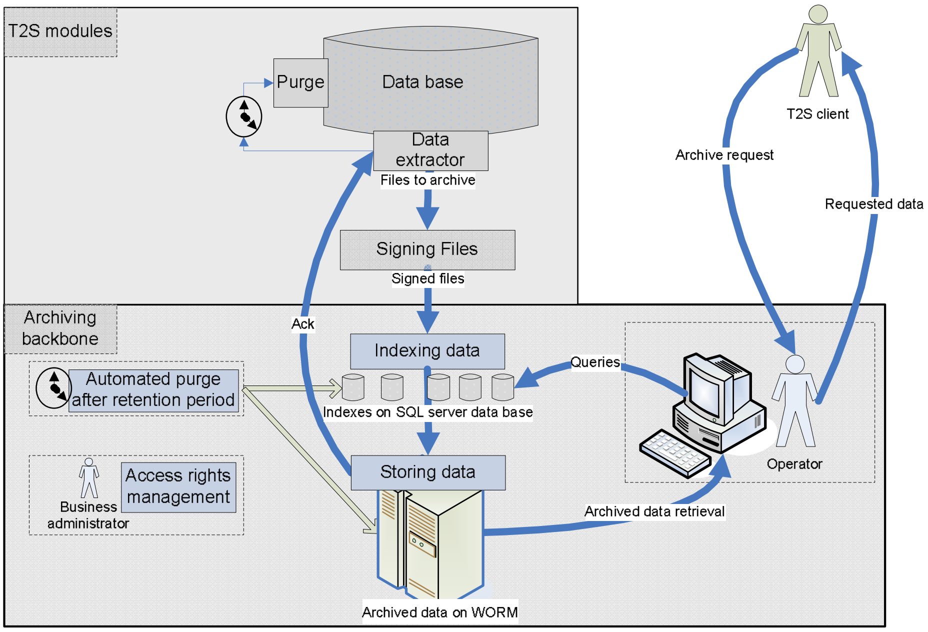


Figure 18 – Legal Archiving processing principles

Capture

The capture is the STS.Net phase which corresponds to the acquisition of signed files and metadata. It begins after file transfer from the client applications (T2S modules). At this level the distinction of the various document types resulting from the same data stream is foreseen.

Querying

The queries are made via a request on metadata (indexes). This request is passed on via HTTP to the STS web server. The server looks for the corresponding entries in the indexes database, which constitutes the repository of the archived objects.

Displaying

Next to querying, the document searched is selected in the list. The corresponding object within CENTERA bay is then retrieved and returned in a HTTP stream where it is shown in a viewer.

Retrieving

This process consists of extracting from the archiving platform all the documents which fulfil the criterion previously filled in. The possibility of executing this command is subjected to privilege. It allows the applications customers to get back their data in their state of origin with the aim of application treatment.

2.2.10.2. Internal technical components

The development of the application STS WEB is realised in a Windows IIS environment. It relies on a Web application running on an application server: the principle is to centralise the execution of the STS applications on a replicated server.

Software architecture distinguishes clearly the various applicative layers constituting the application (presentation, logic, treatments, access to the data). This approach allows, among others, to respect the rhythm of evolution of every stratum, without questioning the integrity of the others, and guaranty a smooth scalability of the solution.

STS applications

One STS.Net application is deployed for each client application (i.e. for each T2S module).

Confidentiality of the data is guaranteed by the management of the privileges within each STS.Net. Every application is given its own pool of data on the CENTERA bays, its own indexes databases, its own rights management and functional administrator.

CENTERA bays

Production and test environments are segregated within the CENTERA bay.
Moreover a dedicated and protected area, named pool, is defined for each client application.

The organisation of archived data within CENTERA bay is described below:

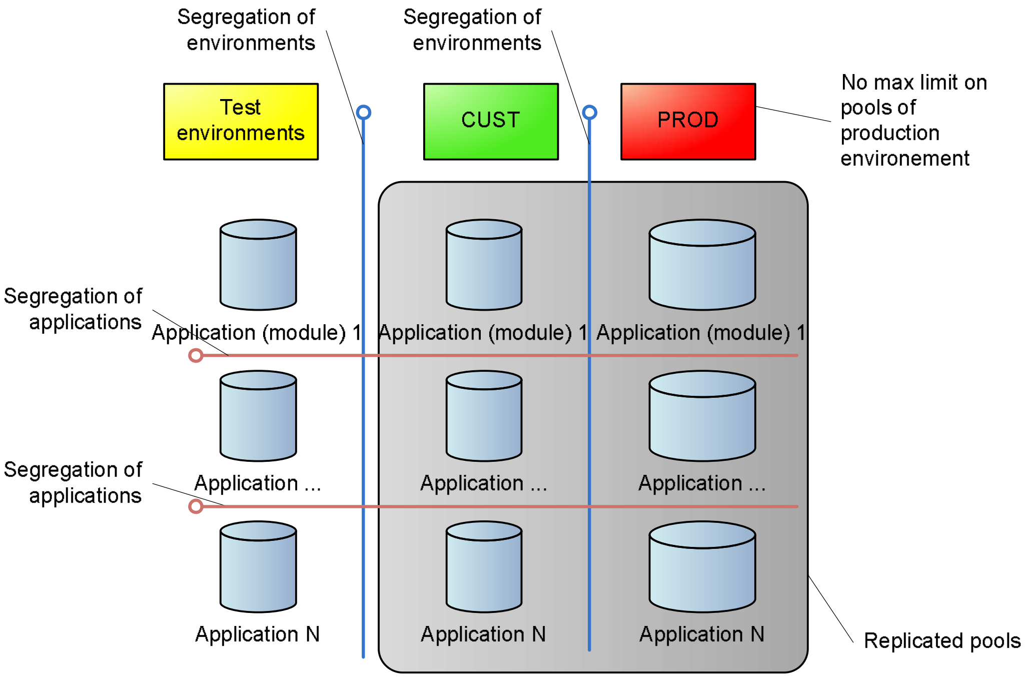


Figure 19 – Organisation of data

Maximum limits are defined on pools, except production, so as to avoid any overflow from the tests environments. The max limit of each pool can be adjusted on the fly, either for increase or decrease, and automated monitoring being done on both disk usage and size of pools.

2.2.11. Scheduling

2.2.11.1. Architecture overview

The following diagram represents the software objects implementing the Scheduling module functionalities, the database and the main internal and external data flows. Of course the software objects are identified starting from the functionalities, nevertheless there is not a relationship 1:1 between functions and software objects.

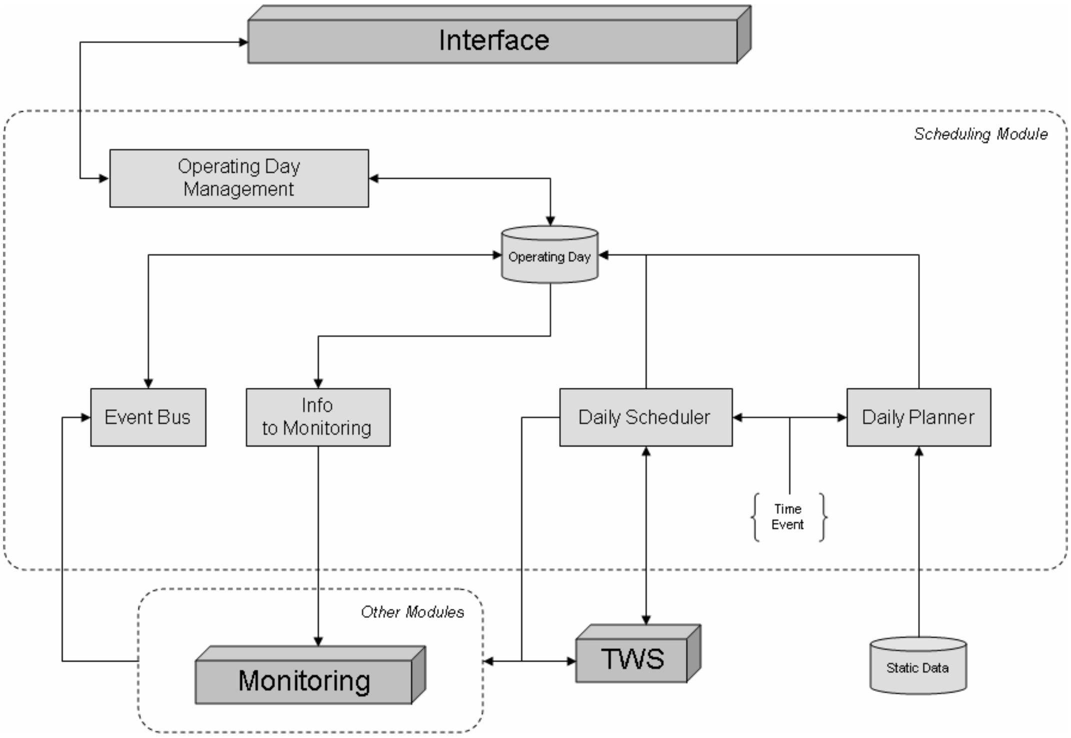


Figure 20 – Scheduling architecture overview

The purpose of the Scheduling module is to manage operating day events and related business processes, with the aim to integrate domains/modules needs, TWS services and Automation services.

On the base of the Static Data relevant information, at Start of Day, the Daily Planner loads the Operating Day database with the current date schedule. At the right time the Daily Scheduler process triggers the relevant other domains/modules either directly or via Tivoli Workload Scheduler (TWS).

The users can interactively change the current date schedule (Operating Day Management process). The other domains/modules can insert immediate execution events (via Event Bus process), in order to communicate events (connected with the business date flow) to the other domains/modules.

2.2.11.2. Processing principles

The module distributes events[7] and reacts on events from the outside (e.g. from other modules or functions). These messages based on IMC/DC queues and WebSphere MQ product.

Each triggering events can wait for feedback or, otherwise, act in a "fire & forget" mode (e.g. for broadcast messages).

The Scheduling module is implemented using COBOL language and uses, for transactional and messaging services, IMS/DC and WebSphere MQ.

The database is owned by the module and is "encapsulated" into the services provided by the module. Read only access to the database is realised through synchronous interfaces (developed as DB2 stored procedures, also written in COBOL). Services which modify the database are asynchronous components developed as IMS transactions (using also WebSphere MQ messaging).

The module foresees a connection to the products Tivoli Workload Scheduler (TWS) and NetView.

Scheduling module also includes also features to provide information to Monitoring about the technical and business events (but not transactional events).

2.2.11.3. Data

The module owns its private[8] database. This database contains information about the Business Date of the current Operating Day and detailed information about the events schedule for the current Operating Day plan and for all the previous Operating Days plans (e.g. schedule time, start time, end time). The new Business Date when Operating Day changes is determined using Calendar information from the Static Data Module. The events schedule for the new Operating Day at Business Date change is determined using Default Events Schedule information from the Static Data Module.

2.2.11.4. Internal technical interfaces

The module is able to provide the Current Business Date to the other domains/modules with a synchronous interface.

The module can activate the TWS in asynchronous mode.

The module can be integrated with TWS for some specific features or to realise complex processes.

In principle, the module interfaces the other domains/modules in asynchronous mode. The technical connections to the other modules is realised as described in chapter 2.1.2.3 "Internal technical Interfaces".

[7] Not transactional events such as, for instance, the validation of a settlement instruction or liquidity transfer orders etc.

[8] I.e. only the Scheduling can read or update the Scheduling database. The other T2S domains/modules can only ask services to the module.

2.2.12. Operational Monitoring

2.2.12.1. Architecture overview

The following diagram represents the software objects implementing the Operational Monitoring, the database and the main internal and external data flows. Of course the software objects are identified starting from the functionalities, nevertheless there is not a relationship 1:1 between functions and software objects.

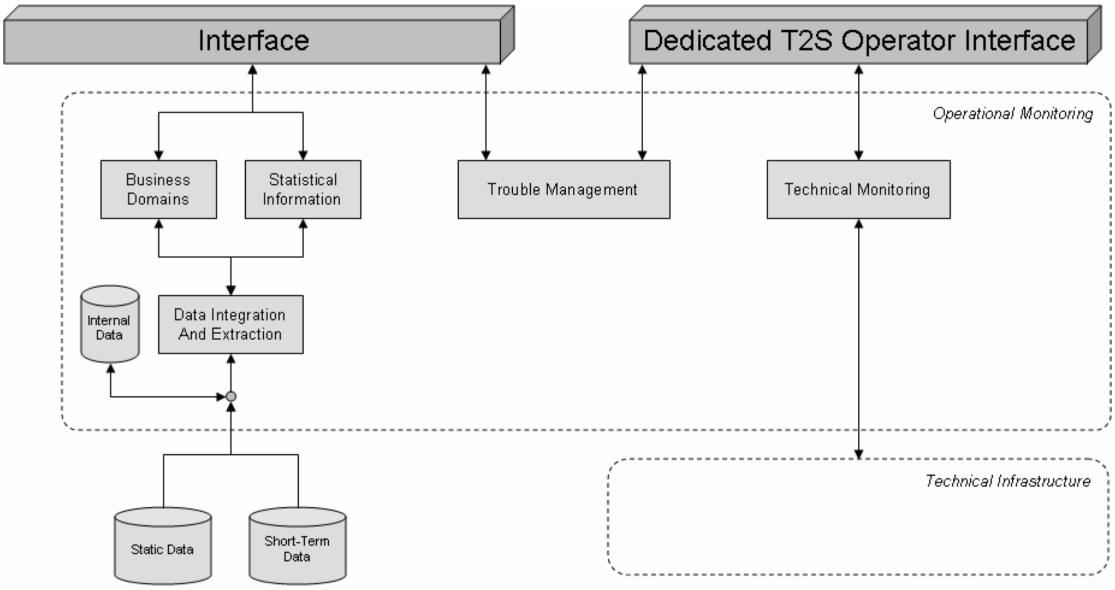


Figure 21 – Operational Monitoring architecture overview

The Operational Monitoring provides support to the T2S Operator in the monitoring of the system, facilitating (i) the detection of functional or operational problems (Business Monitor), (ii) the detection of hardware and software problems (Technical Monitor) and (iii) and the provision of information about the status of an incident/problem and the related analysis and solution ("Trouble Management").

The Trouble Management and the Technical Monitoring are black boxes provided by the Infrastructure. They have private interfaces (even if the Trouble Management can be partially accessed also from the outside, via Interface domain).

The Business Monitor is based on the "Business domains" and "Statistical Information" processes (which manage users' requests) and the "Data Integration And Extraction" process (that integrates the information coming from other domains/modules and provides to the user synoptic views of the business system state).

Data used for BM come from a second database which is updated from the main one by extracting only the information strictly needed for the monitoring in order to avoid negative interdependencies with the productive processes. In addition, some other information is calculated and stored by the module itself processing the operational data (e.g. elapsed time of the settlement instructions for monitoring of SLAs).

2.2.12.2. Processing principles

The Technical Monitoring is realised with an existing product solution (i.e. no special application development). TM for T2S is built starting from T2 architecture, with the necessary adaptations given their differences (e.g. use of Parallel Sysplex) and the progress in the relevant technology.

The Trouble Case Management is realised with an existing product solution (i.e. no special application development), starting from the T2 experience.

2.2.12.3. Data

Data objects

The module relies on:

- Statistical Information module's databases (short term data);
- a mirror database for the static data.

The module has also its own database, with fixed information whose purpose is to customise the module (e.g. threshold alarm).

Data access and storage

The transactions must retrieve data without interfere with the operational data. On this purpose two measures can be used (or a mix of the two):

- use an asynchronous copy of a subset of the operational DB (that grants integrity of data, but suffer a delay of the information);
- read the operational DB, but through "dirty read" (no delay of information, but there is the risk of possible inconsistency of data).

2.2.12.4. Internal technical interfaces

The domains have a set of technical interfaces with the Interface domain. These technical interfaces work in synchronous or asynchronous mode, depending on the type of request/function (query, report, management etc.).

As specified in the "Data Object" paragraph, the module reads directly the mirror copy of the short-term data (in any case the 2 options described in he "Data access and storage" paragraph have to be taken into account);

2.2.13. Data Migration

2.2.13.1. Architecture overview

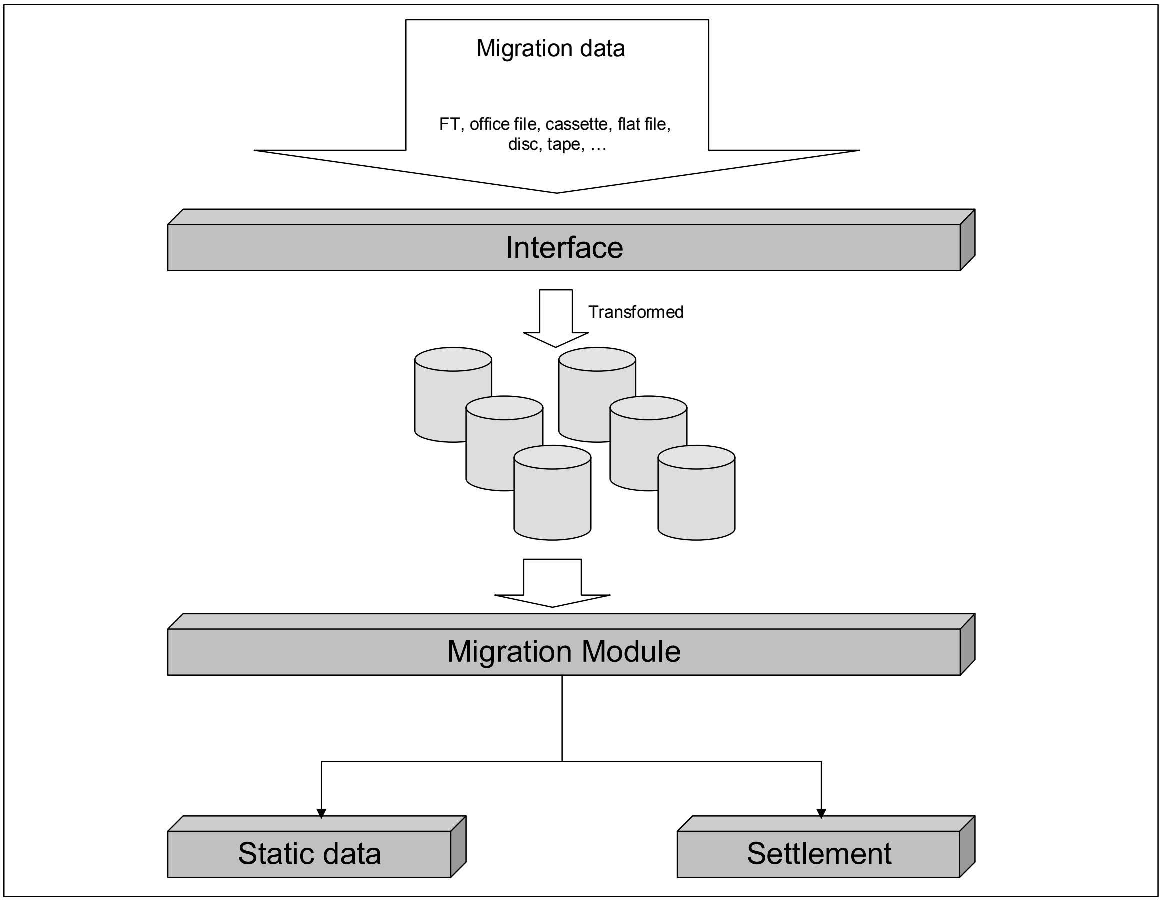


Figure 23 – Data migration architecture overview

The "Data Migration module" is part of the "Operational Services domain". It provides migration functionality to participating CSDs in order to import relevant static and dynamic data from the migrating CSDs into T2S. Thus the T2S system can be initialised without a long manual work of operators entering the data of all CSDs. According to the current design the data is coming from the CSDs as files in the format of flat or MS Excel files. Nevertheless the implementation is kept open to use other formats if requested by the CSDs (e.g .XML format especially since nearly all modern programs like the MS Excel support the export of content as XML files). For the extraction of data from Excel files the help of a library like Apache POI is used.

As the "ICM-T2S domain" is the gateway to the T2S platform all the migration data has to pass this domain and are transformed here into a consistent format that is provided as a mainframe file to the following functions. These functions which are implemented as IMS COBOL programs read the files and process checks for correctness, plausibility and completeness.

2.2.13.2. Processing principles

In order to minimise the efforts for the "Data Migration module" as well as for the other modules it is envisaged to reuse already existing functionalities as far as possible. Included into the module are all features that are used only for the migration process but never during the rest of the business time. As the "ICM-T2S domain" includes all needed interfaces to internal domains also communication related to migration uses the domain as mediator.

2.2.13.3. Workflow

At the start of a migration the T2S system has to be started and initialised. After that the loading of the data can begin. This can only be induced by the CSDs, with the "Data Migration module" waiting for this initial loading.

Firstly all incoming files are checked by the "ICM-T2S domain" concerning role based permission and eventually basic structure and syntax checks. Then they are transformed into mainframe files and the the "Data Migration module" is informed about the availability of the data. The module reads the files and performs checks for correctness, plausibility and completeness. Therefore it stores the data of different files of one CSD in temporary tables. If all checks are successfully passed the data is stored in the respective domain via the access layer provided by that domain. The whole processing is logged in a file or table. There also the reason for a rejection of the data can be found.

2.2.13.4. Data

All files are stored by the "ICM-T2S domain" which also has a functionality to enable the technical team to reload the data of the files manually if some problems occur.

The "Data Migration module" stores the intermediate data in temporary tables for the checks. The result of the validation process is logged as well as the result report.

2.2.13.5. Internal technical interfaces

For the communication between the "ICM-T2S domain" and the "Data Migration module" Websphere MQ (or STP) is used for announcing the availability of files with migration data. For the data itself a serialized file access is employed. That means that firstly the file is written by the "ICM-T2S domain" and afterwards when it is done and closed the "Data Migration module" has access to it. The result reports are sent via Websphere MQ to the "ICM-T2S domain" and forwarded to the CSDs via A2A.

3. Infrastructure Design

3.1. General overview

T2S infrastructure is deployed over three Regions. Two Regions (Region 1 – Banca d'Italia - and Region 2 - Deutsche Bundesbank) host the T2S core business applications (e.g. instructions settlement); the third Region (Banque de France – Region 3) hosts other T2S functions that are not closely coupled with the core business applications (e.g. Legal Archiving).

To allow continuous operations without service interruptions (e.g. in the case of a power outage), each of the 3 regions consists of a primary and a secondary site which run independently from each other. Each site is established as a high availability data-centre for operations (e.g. redundant connections for power supply and use of UPS). This infrastructure has been adopted by the 4CB because it allows T2S offering "state-of-the-art" service levels, notably with respect to security, availability and business continuity.

Regions 1 and 2 are based on zOS, Open systems and Windows technologies, with full back-up and workload distribution, while Region 3 relies on Windows and Open systems.

The main components of the architecture are:

- a central processing system (based on z/OS) hosting T2S core business applications;
- open systems for specialised functions;
- a storage subsystem with synchronous and asynchronous mirroring functionality for efficient business continuity;
- a network interface;
- secure external networks for the connection of CSDs, Credit Institutions, Market Infrastructures and National Central Banks (NCB) to T2S system;
- a dedicated internal network connecting the processing sites (4CBNet);
- security systems.

Multiple independent processing environments have been set up to support development, technical integration, internal and external acceptance, customer testing and live operations (refer to 3.2.2 Logical environments).

Regarding technical operation, T2S functioning relies on automation wherever possible so as to reduce human errors and simplify infrastructure management **{T2S.19.320}**. In addition, T2S offers adequate monitoring tools and security levels (confidentiality, integrity, auditability, identification/authentication and access rights).

The workload of T2S core business applications is distributed between Regions 1 and 2 **{T2S.19.020} {T2S.19.290}**; indeed, while Region 1 is hosting T2S production, Region 2 is hosting T2S Test & Training. Regular swaps ("rotation") ensure proper precaution against regional disaster and keep technical and operational staff skilled in each region **{T2S.20.350}**. Rotation activities in Regions 1 and 2 do not impact systems in Region 3. In order to achieve the technical independency between Target2 Production and T2S Production, the two environments always run on the separate regions in normal operation activity. For instance while T2 Production is running in Italy, T2S Production is running in Germany (or vice-versa); moreover the flows exchanges between the two systems are asynchronous (one region can be active while the other can be stopped). Only in the case of a Regional Disaster (two sites of the same Region are completely unavailable) T2 and T2S Production will be active in the same Region.

The system and the application software in Regions 1 and 2 are kept aligned **{T2S.19.070}** by means of a functionality of the disk storage subsystem, the asynchronous remote copy **{T2S.19.060}**, so that after rotation the system is able to restart with the same customisation (i.e. naming convention, security policies, management rules etc.) .

Like TARGET2, T2S offers its users a single interface, meaning they do not perceive in which Region a certain module is running. Moreover, rotation is fully invisible to CSDs, NCBs, users and market infrastructures, thus no configuration changes in customer systems are envisaged **{T2S.19.090}**.

Even relying on TARGET2 technical and organizational framework whenever possible, a few enhancements are deemed necessary to suit T2S peculiarities:

- Due to the large transaction volumes expected for T2S, it could be necessary to upgrade the current SSP architecture – based on a single image – to a parallel configuration **{T2S.19.180}**: the SSP was designed having in mind the possibility for this kind of configuration (multiple images)[9]. T2S applications will be designed accordingly; the Rotation and Regional Disaster procedures inherited from TARGET2 will be adjusted;
- Communication with external users is enhanced to be compliant with the XML standard and the multiple network providers requirements.

N.B. The present General Technical Design defines the high level technical architecture for T2S based on currently available technology. By 2013, any technical advancement could lead to further investigation and possibly changes in the described architecture.

In line with the concept of service oriented architecture, it is envisaged to select infrastructure products supporting open interfaces **{T2S.19.140}**. Whenever possible, most of the infrastructure

[9] The SSP was based on a single image architecture since this was the easiest and most effective solution given the workload expected for Target2.

components are platform-independent **{T2S.19.120} {T2S.19.130}.** except when T2S requirements (high availability, high scalability and high level security synergies with T2) required the adoption of a mainframe solution.

As far as Legal Archiving is concerned, specialised Windows based components are used to ensure compliance with Legal constraints (see paragraph § for a high level description of the data flow). Files that require to be archived use the more convenient Windows platform: they are sent from MF or Unix DB environment to a unique BdF Gateway Server, via SSL FTP or XFB, here they are sealed (they are unchangeable from this point on….). From the gateway Server data is sent by a BdF file transfer component (or directly) to the file folder on the platform which host the data for the specified period of time (for instance 10 year).

.

3.2. High Level Architecture Design

3.2.1. Architecture description

T2S provides securities settlement services for several actors such as CSDs, NCBs, Credit Institutions as well as for Direct participants: TARGET2 and CCBM2 infrastructures as well as further connected RTGS and CMS have also been considered as "*customers*".

T2S environments can be grouped into 2 logical categories:

- "**Production**", for live operations (one environment only);
- "**Test & Training**", including more environments dedicated to development, testing and acceptance activities.

As mentioned, taking advantage of the synergies between TARGET2 and T2S must not lead to tight and risky dependences between these two critical services, i.e. each service is able to run independently from the other **{T2S.19.030}**.

These T2S environments follow the rotation procedures of TARGET2, having the Production environment running alternately in BdI and BBk six months per year **{T2S.19.095}** (the same rule applies to the T&T environments).

Other less critical applications (for instance legal archiving) do not required the two regions – four sites architecture, according to the user requirements (019.020) and therefore can be hosted in one region only.

Figure shows T2S high level design:

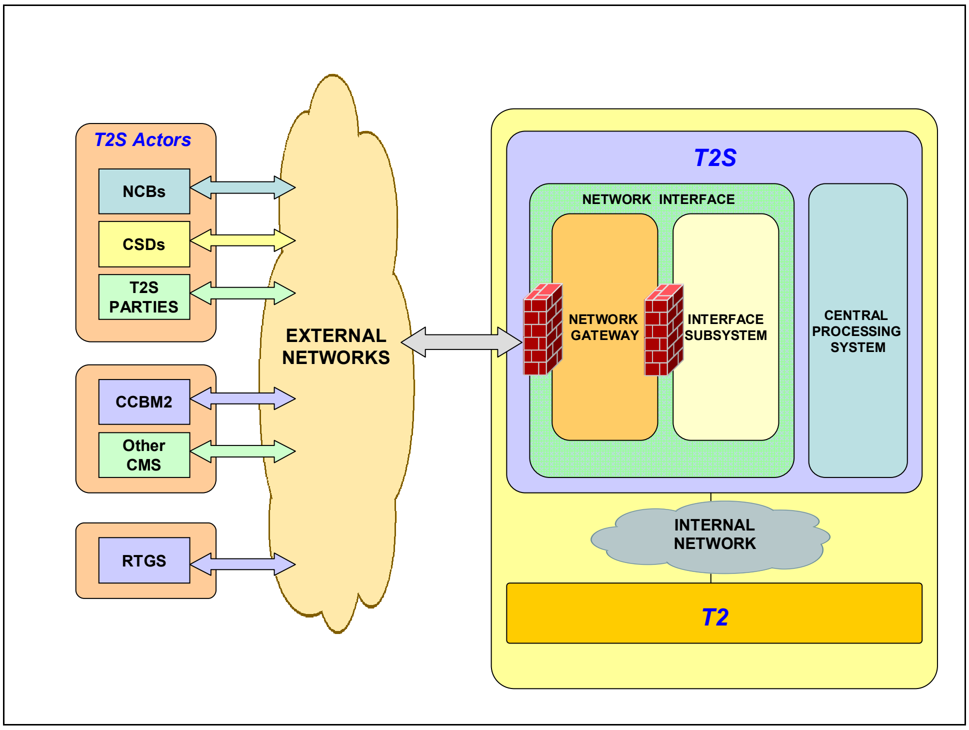


Figure 24 – High level infrastructure design

T2S actors have different connectivity options to access T2S infrastructure (value added Network Providers). Additional services such as access control, load balancing, signature check, etc. are offered by the network providers or by T2S itself.

T2S is connected to TARGET2 via internal network (4CBNet) and mainly using asynchronous communication.

The access to T2S business services is ensured by a dedicated layer named "Network Interface" made up of two components**:**

- a Network Gateway, allowing logical separation between T2S actors and T2S infrastructure;
- an Interface subsystem, general communication layer including middleware providing messaging services for T2S applications.

The implementation of both components is based on open interfaces **{T2S.19.140}** and standard protocols **{T2S.19.150}**.

Operation and management means it is implemented to assure – among the other functionalities – resource monitoring **{T2S.18.480}** and security controls **{T2S.18.550} – {T2S.18.860}**. The clocks of the systems that implement the platform is synchronised **{T2S.18.700}**. The platform provides the means for the segregation – from a logical point of view – of groups of systems/services/information: different broadcast domains are configured at link level (i.e. Virtual Local Area Networks – VLAN) and kept separated by the means of firewalls **{T2S.18.840}**.

3.2.2. Logical environments

T2S requires supplementing the current SSP infrastructure with additional "logical environments" to support development, technical integration, internal acceptance, customer acceptance and live operations. Each logical environment is perceived by users as a working frame in which application functionalities can be run at different stages of the lifecycle.

T2S Test & Training hosts the following permanent environments **{T2S.19.300}**:

- The Driving (DRIV) environment, for testing the first installation and maintenance of the system software;
- The Development (DEV) environment, for the development and unit tests of the application modules. This is the first T2S logical environment and it is shared by the various 4CB teams;
- The Integration (INTEG) environment, used to merge the modules developed by the 4CB and check their compliance with T2S infrastructure. Here the development and the infrastructure teams have the chance to adapt, respectively, the software and the system configurations;
- The Internal Acceptance (IAC) environment, for the acceptance tests;
- The External Acceptance (EAC) environment, allowing users to perform T2S acceptance tests. In this environment, security levels and operational services are the same as in the Production environment. The EAC is also used to test any changes in T2S software;
- The User Test (UTEST) environment, dedicated to customer testing and simulation including testing of CSDs and directly connected instructing parties **{T2S.19.310} {T2S.19.330} {T2S.19.360}**. Its configuration and operations replicate those of the Production environment wherever possible **{T2S.19.340},** yet its capacity and consequently its performance are lower than in the Production environment. The UTEST is used to test customer application changes.

T2S Production includes the Production (PROD) environment for the performance of live operations. This environment is isolated from the Test &Training environments (running in a different region) **{T2S.18.940}.** Hardware and software updates are first applied and checked in the Test &Training environments.

The logical environments are connected to the External Networks and provide different services with possibility of data segregation between testing and production activities. Each environment is connected to the corresponding TARGET2 environment (except for UTEST that was not requested by T2 users) via internal network and mainly using asynchronous communication.

3.3. Main components of Region 1 / Region2

3.3.1. Central Processing System

T2S applications are hosted on a central server using the z/OS Operating System endowed with the following basic components:

- a database management system (DB2);
- a transaction manager (IMS-TM);
- a java application server (WAS);
- a message and queuing server (WMQ).

3.3.1.1. Why Mainframe

In order to meet the strict and demanding requirements on workload and scalability **{T2S.17.010} – {T2S.17.020} – {T2S.17.040} - {T2S.17.050} - {T2S.19.100} - {T2S.19.110} - {T2S.19.180}** a parallel configuration is implemented.

The User Requirements also foresee adaptation of capacity to high workload **{T2S.17.030}**, redundancy against single component failures **{T2S.17.050}** and a high degree of maintainability **{T2S.19.240}**. In this regard, the zOS configuration may be supplemented with the Capacity on Demand (CoD) service.

Based on the above requirements the choice of mainframe is able to support state-of-the-art business continuity solutions and to ensure the requirements specified in the URD are met particularly as regards scalability, availability, reliability and security. In addition, mainframes are nowadays evolving to fully support open standards.

More in detail, the mainframe choice has been based on the following criteria:

- **Powerful Architecture**: processor implements the z/Architecture, has four cores, and accelerators for cryptography, data compression and decimal floating point arithmetic. Unlike other platforms, it uses dedicated CPUs for I/O operations. These CPUs are built in machine and are not priced feature, also they are not charged into software license cost. The design philosophy is intended to minimize machine instruction execution time.

- **Enhanced Security**: z/OS is Security Certified at Evaluated Assurance Level 4+ (EAL4+). Achieving EAL4+ certification will further enable z/OS to be adopted by governments and government agencies for mission-critical and command-and-control operations. While Z/OS was evaluated under the Common Criteria, at EAL4, augmented by ALC_FLR.1, mainframe hardware received EAL5 Certification which is higher than all other platforms..
- **Higher Flexibility**: Compared with other servers, the mainframe offers a simplified, more automated architecture for activation and deactivation of Capacity on Demand processing. In particular, the machine no longer requires immediate, direct contact with IBM for activation of Capacity Upgrade on Demand (CUoD) features.
- New technology and new application technologies support such as:
  - **Java technology-based Web applications and XML-based** data interchange services with core business database environments can be hosted on Z/OS. The exclusive zOS Application Assist Processor (zAAP) is designed to enable these workloads reduce cost-effectively. There is a complete Java EE support and certification, with significant performance improvements.
  - **Virtualization capabilities:** mainframe is the most sophisticated and consolidated architecture supporting the virtualization and sharing of all resources among different workload and operating systems (Dynamic Partitioning), reducing the number of servers needed.
- **Higher Scalability, Availability and Performance Data Sharing**

A **sysplex** is a collection of z/OS systems that cooperate, using certain hardware and software products, to process workload. It is a clustering technology that can provide near-continuous availability.

A **Parallel Sysplex** is a sysplex that uses multi-system **data-sharing technology**. It allows direct, concurrent read/write access to shared data from all processing nodes (or servers) in the configuration without impacting performance or data integrity.

As a result, work requests that are associated with a single workload, such as business transactions or database queries, can be dynamically distributed for parallel execution on nodes in the Parallel Sysplex based on available processor capacity.

A Parallel Sysplex relies on one or more Coupling Facilities (CFs) which is a specialized server, with memory and special channels, and a built-in operating system, containing

- locking information that is shared among all attached systems;
- cache information (such as for a data base) that is shared among all attached systems;
- data list information that is shared among all attached systems.

This "shared data" approach enables workloads to be dynamically balanced across servers in the Parallel Sysplex cluster or be redirected in case of any partition failure. Other platform do not have any CF like architecture where to share the above data. .

3.3.1.2. System environments

One or more logical environments can be hosted in the same system environment as follows:

Test & Training

- DRIV including the Driving logical environment.[10]
- TEST including the following logical environments:
  - Development (DEV),
  - Integration (INTEG),
  - Internal Acceptance (IAC).
- USER including the following logical environments :
  - External Acceptance (EAC),
  - User Test (UTEST).

[10] It is still to be decided if this environment will rotate or not.

PROD

- PROD including the Production logical environment.

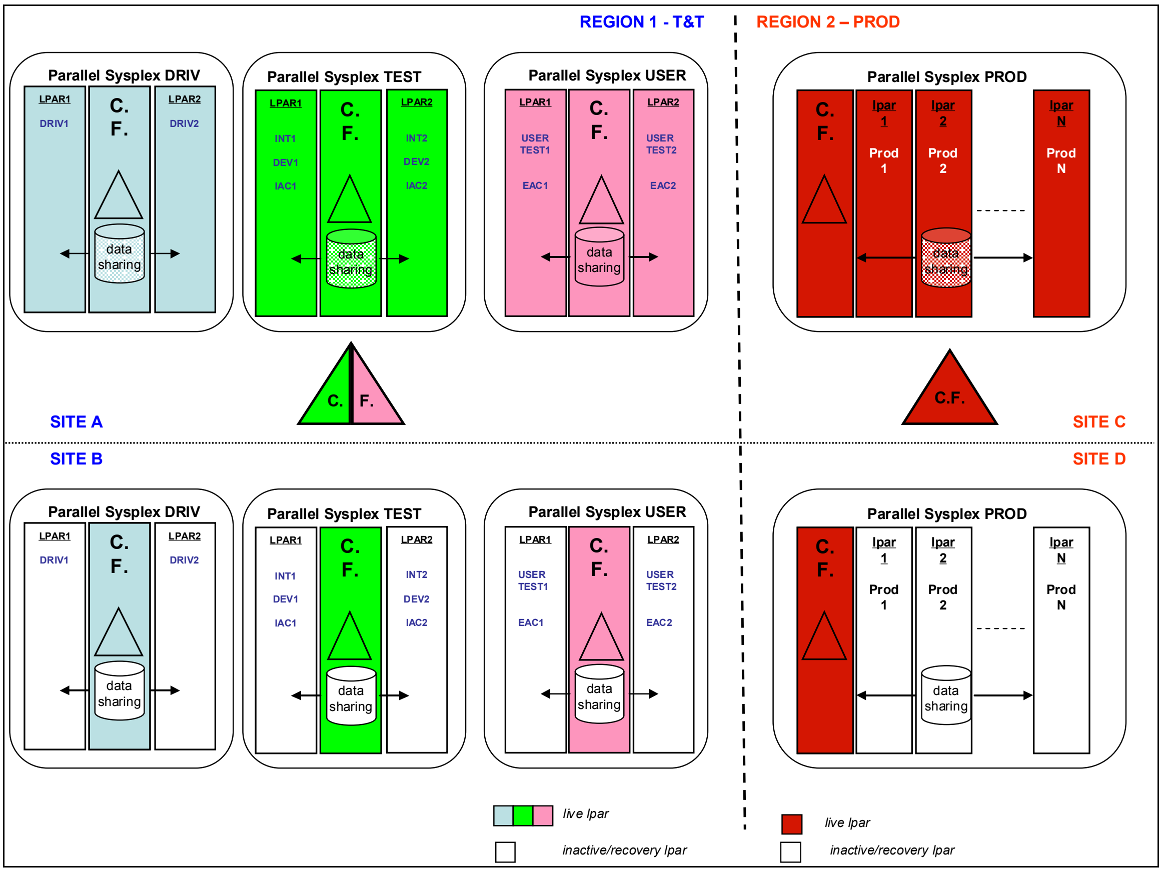


Figure 25 – System environments

3.3.1.3. Parallel Processing Configuration

T2S configuration is based on more Logical Partitions, each hosting the following basic components:

- a database management system (DB2);
- a transaction manager (IMS-TM);
- a Java application server (WAS) ;
- a message and queuing server (WMQ).

The requests coming from the networks are balanced at the central processing system level via a software component (the Sysplex Distributor) based on TCP/IP protocol.

The business logic runs in transaction managers dedicated to Java applications (WAS) and COBOL applications (IMS).

Two products are used to handle the flow from the Sysplex Distributor to the transaction managers: IMS Connect and MQ Series. Both of them provide continuous availability from either an IMS or a WAS perspective. Each IMS has its own IMS queue and processes all the transactions routed to it via IMS Connect or MQSeries.

Figure 26 describes the overall configuration in a Sysplex environment spanned over several LPARs:

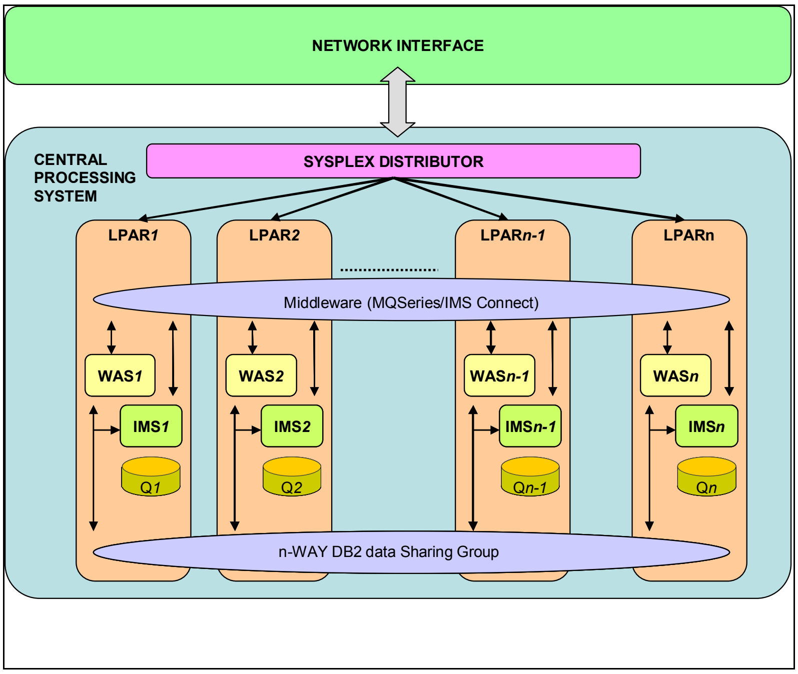


Figure 26 – Workload Balancing for Parallel Processing

3.3.2. Open Systems

In order to guarantee the right degree of interoperability, portability and openness to software standards, T2S infrastructure will widely rely on open systems too. This choice allows the development of an infrastructure independent from proprietary hardware and the implementation of technical solutions based on standards and providing components with a high level of system integration.

The adoption of open systems using the most modern techniques for server consolidation and virtualization, provides at the same time advantages in terms of costs reduction and improved control on the infrastructure by optimising the resource requirements.

T2S infrastructure will make use of multi-partitioned open systems reducing the number of servers to deploy, reducing the costs in terms of software licences and simplifying server administration. Additional benefits are expected in terms of server manageability, obtaining a high level of IT standardisation that helps administrators to better manage their server environment while ensuring high availability and security. Further advantages can be reached in terms of improved service levels and efficiency of operations; an increased product reliability and performance enables higher skilled resources to focus on higher value tasks.

T2S open systems will rely on Unix as preferred operating system; solutions based on Linux distribution will also be taken into account when needed.

T2S open systems will offer a set of specialised services in Regions 1 and 2, including e.g.:

- T2S infrastructure services related to the network interface layer;
- T2S internal DMZs;
- T2S services for the provision of statistics (based on the short term storage i.e. up to 90 days) in order to complement the operational monitoring tools to properly manage the system;
- Operational Monitoring System for real-time detection of functional or operational problems and monitoring of SLA indicators;
- Technical Monitoring System, based on continuous monitoring of the platform's technical components and an alerting system to highlight any unusual occurrences;
- Trouble Management System (TMS) handling the workflow for any incidents or problems and reporting on the respective status; the on-line access to the tool is provided for reporting purposes to CSDs and T2S parties authorized by CSDs ;

To summarize the Architecture Design combines several of the leading technologies and platforms available on the market taking into account the user requirements and best practices,

3.3.3. Storage Subsystems

3.3.3.1. Introduction

As envisaged in the user requirements, T2S storage infrastructure follows the "2 regions-4 sites architecture" **{T2S.19.020} – {T2S.19.290} – {T2S.20.370}**. As a result, it is installed in Italy (IT) and Germany (DE), on the four sites already used for SSP storage infrastructure.

The intra-region sites are identified as follows: for the IT region, Rome1 (RM1) and Rome2 (RM2); for the DE region, Frankfurt1 (FM1) and Frankfurt2 (FM2).

The sites in each Region are geographically separated (>10 km) **{T2S.19.025}** but connected through a DWDM (Dense Wave Division Multiplexer) enabling the use of synchronous communication protocols between the sites (e.g. for data replication).

The infrastructure includes the following components:

- Storage Area Network (SAN);
- Disk subsystems;

- Tape subsystems.

The architecture design and sizing of each component are outlined in the following chapters.

3.3.3.2. Storage Area Network

A geographically extended SAN connecting the 4 sites has been developed for T2S on the basis of the following guidelines:

- T2S storage infrastructure must be completely separated from the domestic storage infrastructure of BdI and BBk, notwithstanding the possibility for synergies with TARGET2 **{T2S.19.010} – {T2S.19.030}**;
- All components of the storage infrastructure must be installed and configured following the same standards on the 4 sites **{T2S.19.030} – {T2S.19.040}**;
- Protection against failures of basic supply for the Data Centre infrastructure requires redundant connection to power and a connection to uninterruptible power supply (as all devices on the SSP) **{T2S.19.050} – {T2S.19.040};**

The Storage Area Network provides basic connectivity between z/OS/Open Systems and storage subsystems (disk and tape) **{T2S.19.100}**, in detail:

It is also used for

- Intra-region Synchronous replication for disk subsystems;
- Inter-region Replication for both disk and tape subsystems.

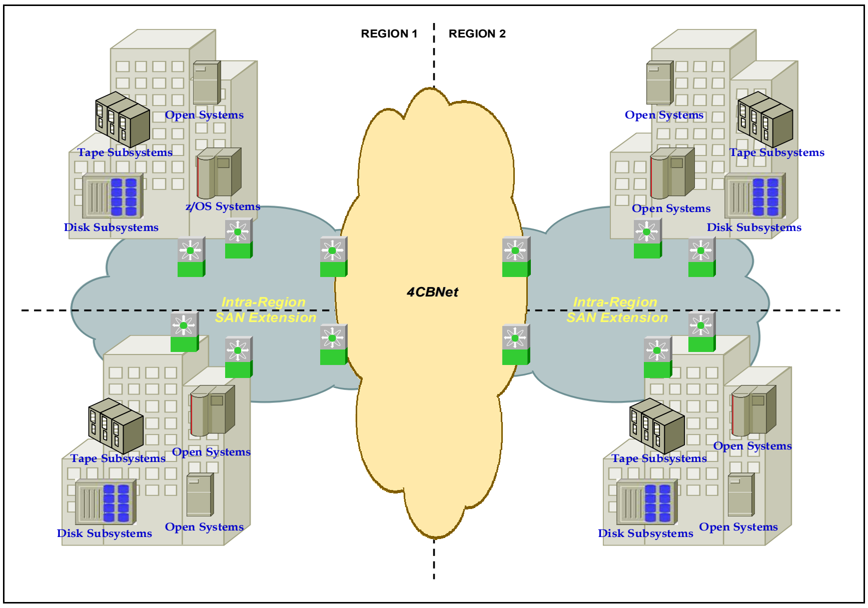


Figure 27 – T2S SAN

Consequently, the architecture provides the following types of storage connectivity:

- FICON for z/OS systems;
- Fiber Channel (FC) for Open systems;
- Fiber Channel for Synchronous replication;
- InterSwitch Links (ISL) for intra-region communication;
- Gigabit Ethernet for Intra-region SAN extension and Asynchronous replication.

In each region, connectivity between the two data centres relies on fibre optics and Dense Wave Division Multiplexer (DWDM) device.

Due to performance constraints and microcode compatibility between Disk and Tape features support and certification, independent SAN directors are recommended for each of the two type of storage. In order to implement a mirrored fabric structure on each site, the T2S SAN is composed of directors providing solutions for all connectivity needs. In detail:

- dedicated directors for tape connection supporting:
    - Gigabit Ethernet connections for inter-region linking;
    - FICON and FC interface for host connectivity;
    - local link through DWDM for site connection within the regions.
- directors for the remaining types of connection (FICON, FC and synchronous replication) supporting:
    - FICON and FC interface for host connectivity;
    - FC interfaces for synchronous copy;
    - local link through DWDM for site connection within the regions.

3.3.3.3. Disk subsystems Design

The basic constraint related to the disk subsystem design lies in the huge disk space and performance requirements.

Two elements have to be considered in defining the number of disk subsystems to be installed on each site:

- Business continuity requirements: these specify that RPO = 0 (Recovery Point Object), wherefore it is most suitable having a single disk subsystem on each site;
- I/O Rate: in the event the foreseen I/O Rate for T2S environment is higher than 100.000 I/O per second, it is preferable to split I/O activities using two disk subsystems on each site (given the current technology).

The present proposal is based on a unique disk subsystem on each site and follows these requirements:

- **Expandability of disk subsystems {T2S.17.020} {T2S.17.030}**. Disk subsystems are fully scalable and thus able to increase disk space for handling the expected growth in transaction volumes in the period 2013-2019;
- **RAID 5 protection** is used in each box;
- **Cache memory** will be used at each disk subsystem for increasing the performance

3.3.3.4. Tape Subsystems

Tape subsystems are used for local backup in Regions 1 and 2 and assure the remote-copying of data between them. They support both z/OS and open system platforms.

Like in TARGET2,

- T2S tape libraries are partitioned to support z/OS and open system backup and archiving activity;
- The z/OS solution is based on **Virtual Tape Server** that makes the Tape Library availability transparent to the system (dedicated hardware device). Tape virtualization allows a host application to access a logical tape drive while the **Virtual Tape Server** actually accesses the physical tape drives;
- Open systems use a specific software tool to address the Tape Library directly.

The existing tape subsystems architecture is implemented as follows:

- 1 partitioned tape library (shared between PROD and T&T) in each region allowing a 2-copy model (1 local copy, 1 remote copy).[11]
- 2 Virtual Tape Servers in each region allowing :
- Load balancing
- data replication among Virtual Tape Servers for business continuity
- Complete independence of Production and Testing environments using the partitioned Tape Library

In T2S, the Tape Libraries are installed on the primary site of Regions 1 and 2, in order to assure load balancing and fault tolerance at the same time; Virtual Tape Servers are installed in each site.

[11] Using partitioned Tape libraries allows a 4-copy model with a tape library installed in each site. The main drawbacks of such solution are related to the additional network bandwidth availability

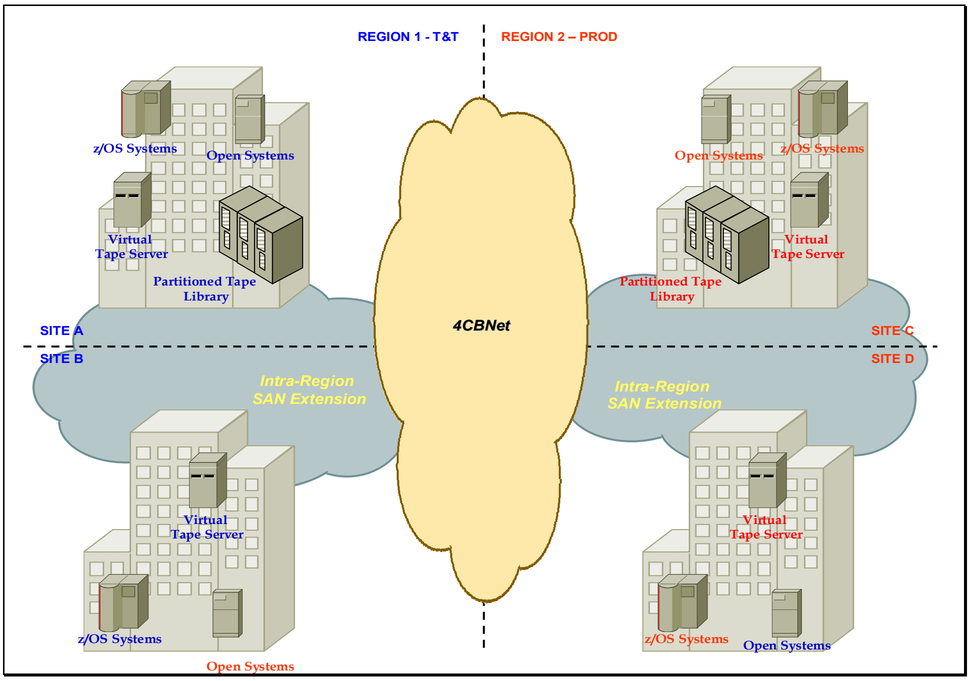


Figure 28 – Tape Subsystems

In a regional disaster scenario, the PROD and TEST environments are up and running at the same time and on the same site (during the restart). Therefore to maintain data segregation between PROD and TEST a partitioned Virtual Tape Server is used.

Due to the long project phase, improvements in technology are expected which could lead to a tape-less solution.

3.4. Main components of Region 3

3.4.1. Open Systems

T2S Legal Archiving function is hosted on Open Systems in Region 3 (Banque de France). The technical architecture for the T2S archiving function will be largely similar to the one adopted for TARGET2 and will allow the 4CB to take advantage of the previous experience and of the acquired skills. Legal Archiving delivers a mechanism to extract and store all relevant archiving data, to retrieve data on user request and purge archived data after its expiration period.

T2S data (transactions and static data) **{T2S.17.080}** is to be archived three months after the related business day **{T2S.17.090}**. The detailed content of such data and the archiving periods **{T2S.17.110}** depend on the CSDs, their home countries and the respective legal framework and are not yet clear. As a consequence, the present architectural design follows a flexible approach avoiding too much detail at this stage of the project **{T2S.17.070}**.

The requirement of having direct connectivity with CSD is referred only to the relevant module and for massive data transfer (T2S 12.190). It shall be possible for CSDs and directly connected T2S participants to connect to T2S via dedicated lines should they wish to do so (for instance for large traffic volumes). This requirements does not apply to legal archiving traffic data.

In line with the "T2S on T2" concept, Legal Archiving for both services relies on the same kind of infrastructure.

T2S Legal Archiving sub-system is integrated into the BdF shared platform "ARCHV".

A partly dedicated infrastructure has been built, meaning that some components of ARCHV are drawn from the existing SSP infrastructure whereas others are dedicated.

Like in Regions 1 and 2, multiple independent processing environments support development, technical integration, acceptance, customer and live operations. Furthermore, for continuity reasons a backup environment exists for live operations. This leads to the following environments:

- **Production**
  - Some components of this environment are dedicated for T2S
- **Development**
  - In this environment, web services, SQL server and file sharing server are on a unique system
  - This environment is shared between T2S and other applications
- **Integration**
  - In this environment, web services, SQL server and file sharing server are in a unique system
  - This environment is shared between T2S and other applications
- **Internal acceptance**
  - This environment has the same characteristics as the production environment
  - Some components of this environment are dedicated for T2S
- **Recovery**
  - This environment has the same characteristics as the production environment.

After go-live, all environments will be used further on for the maintenance of the system.

Access of T2S participants to the Legal Archiving system will only take place through the Network Interface (i.e. no direct access).

3.4.2. Storage

Like in Regions 1 and 2, both primary and secondary site of this Region rely on a Storage Area Network (SAN) and on a set of disk and tape subsystems.

SAN connectivity between the sites is provided by means of fiber optic links and DWDM multiplexers. All components of the storage infrastructure are installed and configured following the same standards in the two sites. This includes redundant connection to power supply and use of UPSs.

Concerning the Legal Archiving function, two kinds of storage infrastructure are used:

- The above mentioned SAN will store archiving rules (meta-data). These are used for the retrieval of the archived data and are managed in a database management system;
- A storage control unit, connected in IP to a front end server via a dedicated VLAN, will store the archived data. The Microcode makes it possible to erase the archived objects only after the expiration of their retention period specified in the metadata definitions.

Region 3 also hosts on its Open systems T2S services for the provision of statistical reports (based on the long term storage i.e. more than 90 days) to be offered to T2S users on optional basis. These components will lean as much as possible on the existing TARGET2 infrastructure.

3.5. Connectivity Services

3.5.1. The User Requirements

The T2S User Requirements define a number of aspects related to connectivity services, which can be summarised as follows:

- Catalogue of connectivity services: A catalogue of connectivity services shall be developed as part of the T2S overall service catalogue. The content of the connectivity service catalogue shall include the network providers offering connectivity to T2S and the services offered by these providers. These shall include dedicated connectivity solutions, backup and scalability.
- Communication: T2S connectivity should cater for online (U2A) and asynchronous modes (A2A) of operation. T2S connectivity services shall support guaranteed delivery, even if the receiver is off-line, and real-time file transfers. These services shall operate in both push and pull mode for both files and single messages. The services will be part of a network tender which is envisaged to select the network providers for T2S. **{T2S.12.260}**
- Architecture: T2S shall not be dependent on a particular technology and shall use standard communication protocols. The connectivity services catalogue needs to be compatible with

T2S's high level of resilience and scalability, and should support its two regions (business continuity) seamlessly for the users.

- Security controls: T2S shall offer appropriate controls and security features. These include technical (identification of communicating actors, authorisation, message protection and routing control) and organisational measures (network services agreement and policy).

3.5.2. Catalogue of Connectivity Services

All directly connected T2S Actors should be able to access the T2S platform via the connectivity provided by (value-added) Network Providers.

The T2S catalogue of connectivity services **{T2S.12.280}** will list the network providers offering connectivity to T2S and the services offered by these providers. These include dedicated connectivity **{T2S.12.290}** solutions and back-up **{T2S.12.320}**. Internet will be a service included in the catalogue, although – unlike the other services - this alternative is for low volume use only and there cannot be any commitment on the quality of the message/file transport.

All T2S actors, including central banks managing connected RTGS and Collateral Management Systems, can choose their preferred connection type(s) from the T2S catalogue of connectivity services and will be fully responsible for their choices.

In order for T2S to be managed effectively, all directly connected actors will be required to exchange information about their intended approach to network connection with the Service Operator (4CB) in advance.

3.5.2.1. Value-added network services

Although no compelling need can be identified for T2S to take any commitment on connectivity to T2S beyond its own network access point within its data centres, the Eurosystem recognizes that network connectivity has an impact on the overall T2S service and on business continuity. In case of an outage in the network or in the interface, the user may perceive this as a T2S service unavailability. Furthermore, inter-region rotation and regional disaster recovery both require the network providers to implement special arrangements to make them transparent to the users. For these reasons, the specification by T2S of an appropriate set of value-added network services is very important. These services are supplied in part by the network providers, and in part by specific arrangements on the T2S platform.

3.5.2.2. Value-added security services

T2S shall ensure the confidentiality, authenticity and integrity of the traffic in transit and guarantees the authentication of end-devices **{T2S.18.360} {T2S.18.820} {T2S.18.850}**.

For this purpose, the "basic" value-added network services are defined as follows **{T2S.12.330}**:

- closed user group management;
- access control to guarantee connections for allowed users only **{T2S.18.810}**, User authentication for external connections;
- non-repudiation of sent messages;

3.5.2.3. Value-added communication services

The communication services provided by network providers are U2A (User to Application) and A2A (Application to Application). A2A supports both guaranteed delivery, even if the receiver is off-line, and real time transmission, it operates in push and pull mode for both files and messages transfers **{T2S.12.260}**. Each network provider will be required offer all these message patterns for communication between directly connected actors and the T2S platform.

They are expected to scale up or down according to central T2S application requirements. Security, resilience and coverage objectives will be included in the SLA **{T2S.18.540}.**

3.5.3. Selection of network providers

The Eurosystem envisages selecting network providers for T2S according to a "licensing" approach. The selected network providers would be allowed – by being awarded a license – to offer their services directly to the T2S Actors, without any commercial involvement from the Eurosystem. In order to be licensed, the providers will have to fulfil a number of technical and operational selection criteria, and commit to certain service levels and liability provisions.

In order to allow the Eurosystem to make an objective ranking so that it can eliminate – if need be – one or more network providers which comply with the standards noted above, those who fulfil the technical, operational and service level criteria will be requested to submit a bid for a license. The term of the licence is for future decision; it is unlikely to exceed 7 years.

As part of this bid, they will agree to reimburse the Eurosystem for all costs related to the connection of their network to T2S. The amounts paid by the winning providers (i.e. both the reimbursement of costs and the license fees) will reduce the costs to be recovered by the Eurosystem via T2S fees. It is also possible that the Eurosystem will require a non-refundable deposit from network providers to cover the costs of detailed due diligence in relation to their technical compliance.

Regarding the number of providers that should be allowed, the minimum will be defined by the desire to allow sufficient competition, while the maximum is defined by the need to manage project and operational risk. The Eurosystem considers that up to 3 providers delivers the optimal balance between the two conflicting objectives. (The internet option – for low volume only – is an additional channel.) Nevertheless, at any point in time, depending on the circumstances, the Eurosystem can decide to offer additional licenses.

3.5.4. T2S architecture

Integration of several Value-added Networks

At its go-live, the T2S platform will be connected to the VANs, containing the services described in section 3.5.3.1 and 3.5.3.2 above.

The following picture shows the foreseen high level architecture.

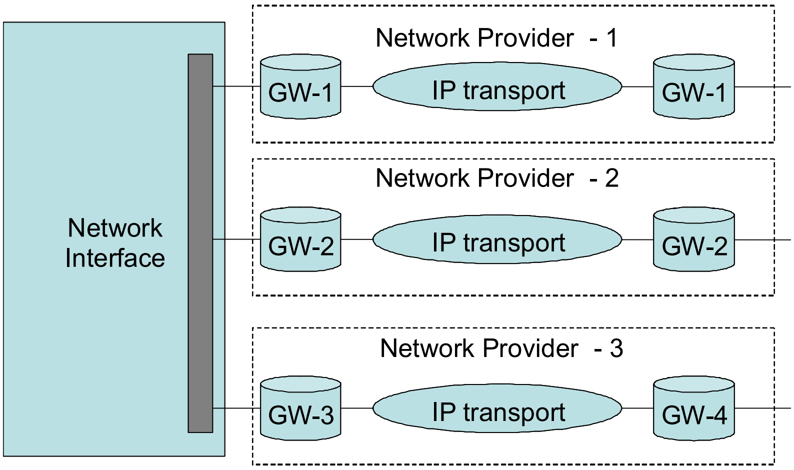


Figure 29 - External connectivity architecture.

The **Network Interface** is within the T2S platform as shown in the above picture. It is responsible for:

- connection to different Network Providers;
- routing the message/file traffic between T2S application and connected networks;
- coping with T2S User Identities, addressing schemas, PKI mechanisms defined (potentially) differently in each interconnected VAN (partner management).

The following picture highlights the functions of the multi-provider interconnection component inside the Network Interface of the T2S platform.

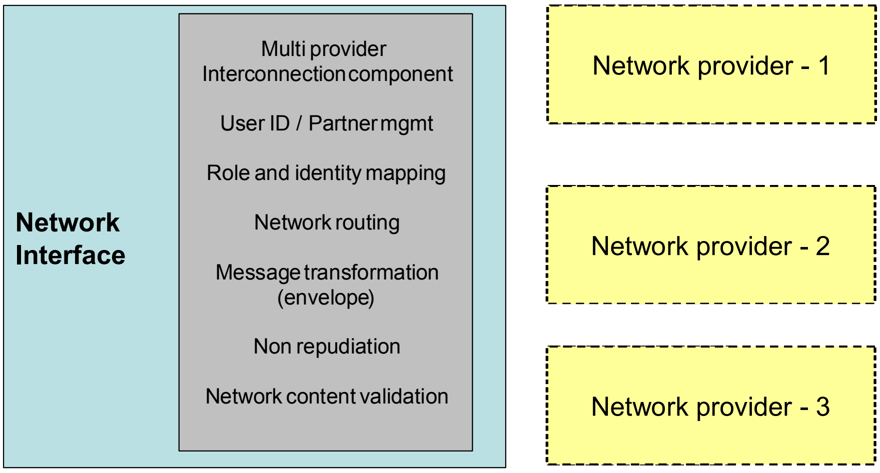


Figure 30 - Multi provider interconnection functions.

Each of the four T2S processing sites in Regions 1 and 2 is interconnected to the backbones of each of the selected networks, ensuring recovery in case of outage or disaster **{T2S.18.360} {T2S.19.020} {T2S.19.040} {T2S.19.050}**. T2S offers the directly connected T2S Actors a single interface **{T2S.19.080}**, i.e. the Rotation of the Interface (between BBK and BdI) is transparent to the T2S Actors and relies on specific network provider arrangements (e.g. changes in the T2S Domain Name System external interface or in the Domain Name System of the service provider **{T2S.19.090}**).

3.5.5. Network Interface

The T2S Network Interface is a set of hardware and software elements allowing interactions between the directly connected T2S Actors (CSDs, NCBs, T2S parties, 4CB, ECB, including CCBM2, TARGET2, other cash settlement (typically RTGS) and other collateral management systems)**{T2S.12.010} {T2S.12.350}** and the relevant T2S modules, both for inbound and outbound communication **{T2S.12.030}**. The Network Interface is the functional layer directly attached to the External Networks **{T2S.12.060}** and consists of two basic components, the Network Gateway (external, and direct front-end to the External Networks) and the Interface Subsystem (internal).

From a logical point of view, each access via External Networks passes through the Network Gateway and Interface Subsystem layers.

The T2S Network Interface handles communication in browser-based U2A mode and in A2A mode. Standards in use are HTTP/HTTPS for U2A interactions, and XML for A2A mode. All A2A communications established through the Network Interface are compliant with the ISO20022/UNIFI standard **{T2S.12.040} {T2S.12.340} {T2S.12.360}**, and also with Giovannini protocol recommendations **{T2S.12.050}** in the particular case of file transfers.

The Network Gateway is directly linked to the physical networks and is intended to increase the security of the T2S system by operating a logical separation between participant and infrastructure. It is devoted mainly to the implementation of controls regarding user identity **{T2S.18.810}**, XML message content validation, protection against malicious code **{T2S.18.500}**. Once checked and validated, the requests are forwarded to the Interface Subsystem layer.

The Interface Subsystem is responsible for making the business logic independent from the various network protocols. Thereby A2A related messages (inbound and outbound) are processed through a message broker acting as an Enterprise Service Bus (ESB) by supporting a broad range of multiple transport protocols and data formats (enhanced SOA approach with advanced Web Services support).

The following picture depicts the Network Interface Architecture and the specific functions related to each component.

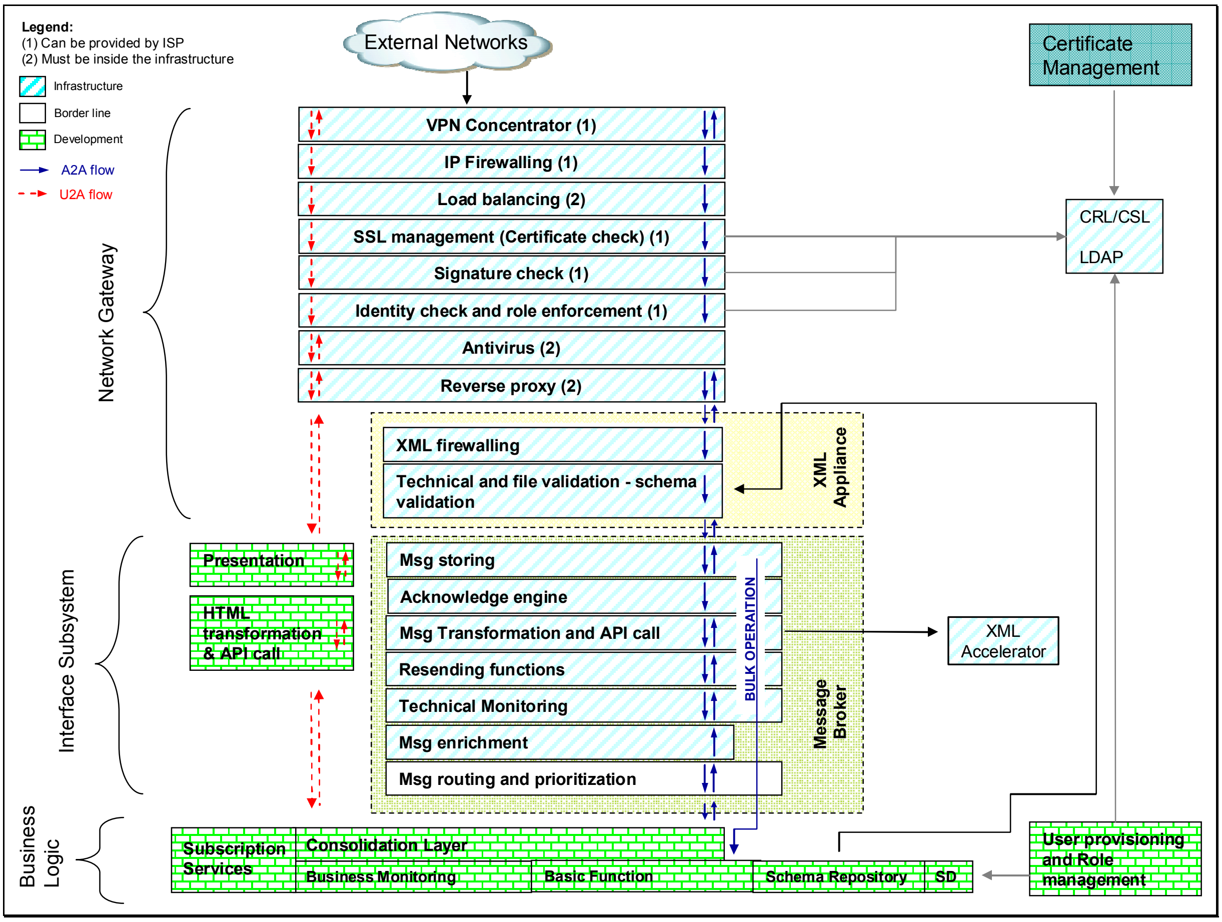


Figure 31 – Network Interface Architecture.

The picture shows two main flows: U2A and A2A. The path followed by each request through the Network Gateway and the Interface Subsystem varies according to the kind of interaction (U2A or A2A) and its direction (inbound or outbound).

XML management through the various layers can be implemented with a combination of software products and hardware appliances taking into consideration the following parameters:

- Performance requirements;
- Message throughput requirements {**T2S.12.160**};
- Message size;
- Additional services needed such as traffic monitoring {**T2S.17.180**}, message editing etc.

The envisaged configuration makes use of XML appliances for XML firewalling and to accelerate technical file validation and transformation, whereas software products (e.g. a message broker) handle the XML message exchange and additional services like flow monitoring and routing functions.

3.5.5.1. Network Gateway

The Network Gateway is designed with the aim to address the requirements related to:

- **VPN (Virtual Private Network) Concentrator, IP firewalling and load balancing.** These components provide a secure communication channel even when the underlying system and network infrastructure are not secure. Sensitive data and passwords should be encrypted against unintended disclosure or modification **{T2S.18.630}**. As connection links are VPN based, a VPN concentrator is needed to manage the connection between the partners and to verify the certificates used by network infrastructure servers. IP traffic is also filtered through a firewall blocking and discarding malicious IP packets. High availability and scalability are exploited by a load balancing function that dispatches the workload among parallel server instances.
- **SSL (Secure Socket Layer) management, signature check, User Identification and Access Management {T2S.12.110} – {T2S.12.120} – {T2S.18.1000}.** These elements identify the sender of the communication (using an identity or signature certificate), check that the communication is received from a secure and recognized technical address, prevent intrusion and unauthorised access to the private network. Security services as CRL (Certificate Revocation List) proxy for certificate validity check and LDAP (Lightweight Directory Access Protocol) server for certificate storing, are provided by an LDAP server and a CRL/CSL (Certificate Suspension List) proxy. In this scheme, the user provisioning mechanism feeds both LDAP server and Static Data.
- **Antivirus**. As U2A flows are driven by HTTP, an antivirus feature is required to avoid malicious code injection.

- **XML firewalling**. A2A is based on XML messages, requiring a specialized firewall to block malicious code based on XML vulnerabilities. This functionality can be provided by an XML appliance.
- **Reverse proxy**. The reverse proxy acts as an intermediary between a user request (U2A and A2A) and back end systems, performing additional checking and logging access control functions as necessary, and connecting to the servers on behalf of the client.

3.5.5.2. Interface Subsystem

The Interface Subsystem provides the following services:

- **Integration of external VANs**
  - adaptation for interconnecting each different VAN
  - routing the message/file traffic among interconnected networks
  - coping with the different authentication/addressing schemas/PKI used in each interconnected light-VAN
- **U2A {T2S.12.250}**
  - **Presentation, HTML (HyperText Markup Language) transformation and API call.** Web Applications are hosted on the presentation layer that manages all the static components (HTML, images, etc.) and performs the operations needed to call the business logic on user request basis.
- **A2A {T2S.12.260}**
  - **Message storing and acknowledge engine {T2S.12.020}**. As soon as the XML message is considered valid and free of malicious code, it can be taken on board, so it is stored and an acknowledge message is generated and sent to the counterparty.
  - **Message transformation, API call**. XML messages may need to be translated from a format into another (e. g. XML to Cobol Copybook) to allow invocation of business logic module (API call).
  - **Resending function and Technical monitoring**. These layers enable resending functions for both inbound and outbound messages for recovery reason or counterparts requests. They provide information related to the occurrence of errors or critical events and data for the production of SLA reports. Data related to the message exchange is also collected at this stage and coupled with the relevant information thus allowing to trace the message/communication flow.

Message enrichment and message routing and prioritization {T2S.12.130}, **{T2S.12.140}**, **{T2S.12.150}**, **{T2S.12.170}, {T2S.12.180}**, **{T2S.12.190}**, **{T2S.12.200}**, **{T2S.12.210}**,

{T2S.12.220}. For the inbound traffic, messages are routed to the interface domain by checking and extracting information in the envelope. Messages are finally forwarded to the business logic Consolidation Layer by the interface domain basing on information inherited by the Interface Subsystem (infrastructure) and payload content. Unbulking operations will also be executed by the Interface domain. For the outbound traffic, messages are routed to the proper network by adding the final address of the message provided by the interface domain. Message enrichment can be performed on interface domain request basis.

3.5.6. Internal Network (4CBNet)

The existing Internal Network (3CBNet), connecting the six operational sites of the SSP in the 3 Regions, and the related DMZ interface **{T2S.19.220},** have been enhanced with network links to provide a stable network connection for Banco de España teams too. This enhanced network is to be considered as a unique telecommunication and security infrastructure to be upgraded and managed as a whole.

3.5.6.1. Traffic flows

Five main types of flows are foreseen:

- storage;
- external network;
- development and file transfer;
- voice and video over IP;
- management, monitoring and internal support.

Storage flow

Storage activities generate the largest traffic flow compared to other activities.

Storage copy is synchronous between the two sites within Region 1 and within Region 2 and asynchronous between Region 1 and Region 2.

The synchronous copy is transported on high speed fiber optic links, provided by Central Banks, inside each region; the asynchronous remote copy is transported geographically through high speed international geographical links.

External network flow

The 4CBNet bears external network traffic, collected by the external networks and addressed to the T2S system.

Development and file transfer traffic

Development and file transfer traffic refers to all kinds of development activities and file transfers from the 4CB domestic environments to the T2S environments and among the four NCBs.

Voice and Video over IP traffic

Network services support voice and video communications among the 4CB members.

Management, monitoring and internal support traffic

All alarms sent by devices and servers and SNMP (Simple Network Management Protocol) traffic are part of this category.

Authorised users on the domestic 4CB will be able to connect to all environments, for management and application development purposes.

3.5.6.2. 4CBNet architecture

The 4CBNet provides IP connection services among the sites in BdI, BBk, BdF and BdE **{T2S.19.020}** with a high level of resilience due to redundant paths between the sites **{T2S.19.040} {T2S.19.050}**.

The network design takes site rotation into account and is adequately robust. For inter-region traffic, the 4CBNet provides twice the maximum bandwidth required, so that even in the event of a single failure any impact on performance can be averted.

For exploiting synergies with TARGET2-SSP infrastructure **{T2S.19.010}**, the network configuration is based on the existing high capacity links connecting Region 1 and Region 2 and the links connecting Region 3; but an adaptation will be done in respect of needed throughput and to connect also Banco de España.

The internal network is designed on the basis of estimated traffic volumes expressed in the URD **{T2S.17.030}**. The use of resources is monitored during operations allowing efficient Capacity management **{T2S.18.480}**.

The 4CBNet is depicted in Figure , which highlights the following functions:

- WAN connectivity among regions;
- Confidentiality, integrity protection of traffic in transit among regions by means of cryptographic techniques supported by key management solutions **{T2S.18.1030}** according to T2S security policies **{T2S.18.1020}**;
- Authentication of WAN links end-point devices **{T2S.18.820}**;
- Switching and routing at each region;

- Firewalling for integration of external interfaces;
- Jumbo Frames support.

So appropriate measures are taken to fulfil the T2S security requirements **{T2S.19.100}**; security objectives and service levels are part of the network service agreement **{T2S.18.540}**.

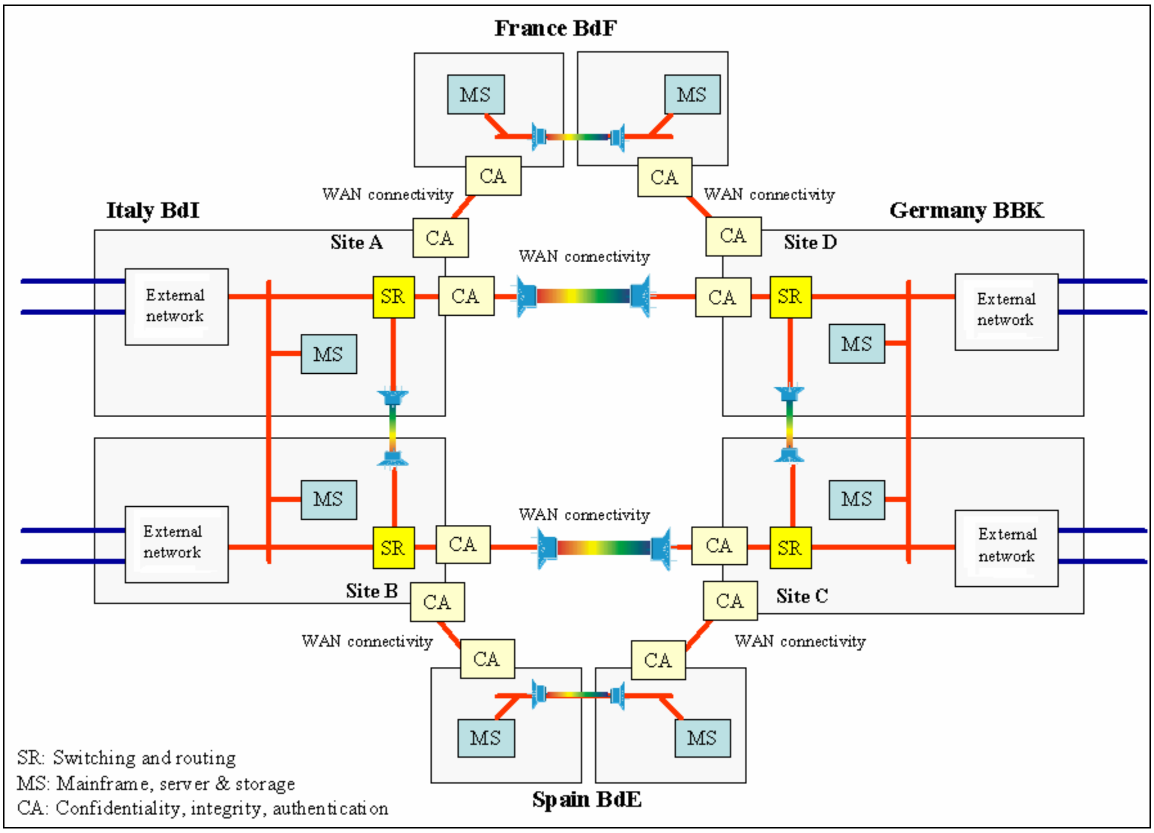


Figure 32 – 4CBNet Design

The needed bandwidth between Rome and Frankfurt is provided by high capacity links, whereby needed scalability is ensured **{T2S.19.200}**. The links with Paris and Madrid have a lower capacity but with similar SLA providing high availability for the whole 4CBNet **{T2S.19.210}**.

Routing information is managed by a dynamic routing protocol, smoothing the rotation (between BBK and BdI) burden and supporting automatic re-routing in case of a failure, like for the 3CBNet **{T2S.19.090}**.

3.6. Business Continuity

3.6.1. Introduction and assumptions

Business Continuity is the ability to adapt and respond to risks in order to maintain continuous business operations. The provision of Business Continuity involves three main aspects:

- **High availability** is the capability to guarantee a service regardless of local failures in business processes, physical facilities and IT hardware or software;
- Allowing **Continuous operations** is the capability to guarantee a service also during scheduled backups or planned maintenance;
- **Disaster Recovery** is the capability to recover a data centre at a different site in the event a disaster destroys the primary site or otherwise makes it inoperable. The distinctive feature of a disaster recovery solution is that processing resumes at a different site and on different hardware.

The Business Continuity proposal for T2S was outlined on the basis of the following guidelines:

- The business continuity model shall be able to respond to the widest possible range of system failures **{T2S.20.390}**;
- The technical environment for T2S core system shall follow the "two regions/four sites" architecture **{T2S.19.020} – {T2S.19.290}**;
- Each of the four sites of T2S must be able to fulfil the agreed service level **{T2S.19.040} – {T2S.20.370)}**;
- Specific procedures have to be defined for rotation between the 2 regions and switch between the 2 sites within the same region **{T2S.20.350} – {T2S.20.360}**;
- Complete logical independence between TARGET2 and T2S operations will always be guaranteed **{T2S.19.030}**;
- Disaster recovery time will be under 2 hours from the moment when the decision is taken **{T2S.20.410}**.

Loss of data and loss of uptime are the two business drivers that serve as baseline requirements for a Business continuity solution. When quantified, they are more formally known as **Recovery Point Objective** (RPO) and **Recovery Time Objective** (RTO) respectively:

- The RPO is a point of consistency to which a user wants to recover or restart. It is measured in the amount of time between the moment when the point of consistency was created or captured and that when the failure occurred;
- The RTO is the maximum amount of time required for recovery or restart to a specified point of consistency.

Along the lines of the "T2S on T2 concept", the business continuity solution for T2S replicates what already is in place on the SSP. Consequently for T2S, three types of service interruptions have been considered:

- **Short continuity failure** is understood as a short service interruption (e.g. due to component failures, a system reboot, or a line failure). These problems may typically be solved at the primary site.

- **Major failure or disaster** is understood as a serious service interruption (e.g. disruptions caused by fire, flood, terrorist attack or major hardware/ telecommunications faults). These events require the activation of an alternative site.
- **Regional disaster** is understood as a "wide-scale regional disruption" causing severe permanent interruption of transportation, telecommunication, power or other critical infrastructure components across a metropolitan or geographical area and its adjacent communities; or resulting in a wide-scale evacuation or inaccessibility of the population within the normal commuting range of the disruption's origin. These events require the activation of an alternative region.

3.6.2. Business continuity design for Region 1 and Region 2

Like in TARGET2, the architecture of T2S core system is based on the concept "2 regions / 4 sites". The four sites are fully equivalent and each of them is equipped with the same technical resources: processor, storage, network interface, software, etc.

The business continuity design involves different levels of the overall architecture requiring various technical solutions. The main elements for the design are:

- **Redundant HW components**. Hardware systems will have redundant components in terms of power supply, memory and processor to assure service continuity also in case of failure of a single component ;
- **Scalable central processing system**. Parallel Sysplex implementation assures dynamic load balancing and additional business continuity in case of failure of a single logical partition (refer to 3.3.1.3 Parallel Processing Configuration);
- **Cluster configuration** for Open Systems running critical applications assures continuous service ;
- **Storage Area Network Extension**. An extended SAN increases the geographic distance allowed for SAN storage operations, in particular for data replication and copy. This is especially relevant for the protection of data in the event of disaster at the primary site;
- **Recovery Technologies**. The following technologies support data recovery with increasing effectiveness (minimization of data loss):
    - Tape backup and restoration;
    - Periodic replication and backups;
    - Asynchronous replication for disk subsystems;
    - Synchronous replication for disk subsystems.

Data replication technologies are employed with two primary objectives:

- To copy data to a recovery site and minimize the RPO;
- To enable rapid restoration (RTO).

Disk-based replication uses modern intelligent storage arrays, which can be equipped with data replication software. Replication is performed transparently to the connected servers (hosts) without additional processing overhead. Replication products can be categorized as synchronous or asynchronous:

- **Asynchronous replication** is a nearly real-time replication method in which the data is replicated to the remote array some time after the write operation is acknowledged as complete to the host. That means application performance is not impacted but loss of replicated data is possible (RPO >0). The enterprise can therefore locate the remote array virtually any distance away from the primary data centre without any impact on performance, but the bigger the distance, the greater the risk of data loss in case of disaster;
- **Synchronous replication** is a real-time replication method in which the data is written to the local storage array and the remote array before the write operation is considered complete or acknowledged to the host (zero data loss or zero RPO).

Synchronous and asynchronous replications allow fulfilling T2S business continuity requirements.

In detail, synchronous and asynchronous replications are active independently and simultaneously for the various T2S environments. Replication is synchronous between primary and secondary site of each region and asynchronous between Region 1 and Region 2. Assuming that the Production environment is running in Region 1 and the Test & Training environments in Region 2, a local synchronous copy (intra-region) and a remote asynchronous copy (inter-region, between BdI and BBK) will be created for each environment.

In order to enable the system to restart on a remote site in the event of disaster, consistency between the two remote copies (synchronous and asynchronous) is guaranteed by the storage infrastructure.

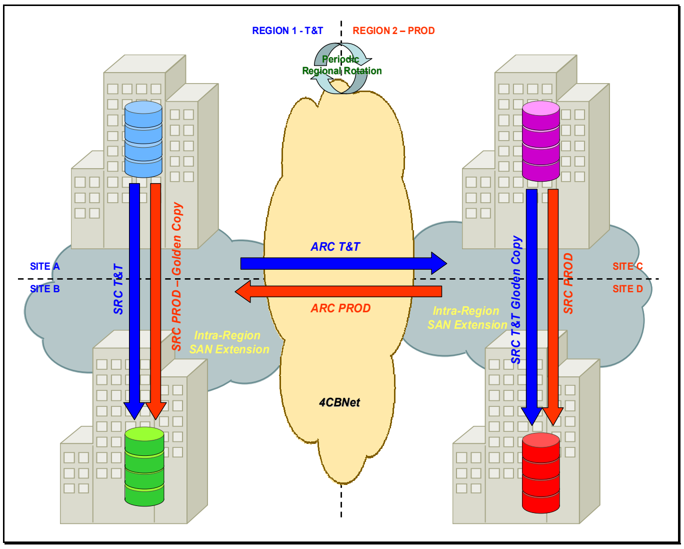


Figure 33 – Business continuity design for disk subsystems

Business continuity is also assured on Tape Subsystems using:

- **SYNC/ASYNC replication between Virtual Tape Server** inside each region. This feature assures that in case of short continuity failure or intra-region recovery, no data loss occurs for Tape subsystems
- **2-copy model**. Assure that data is replicated to the tape library in both the Regions.

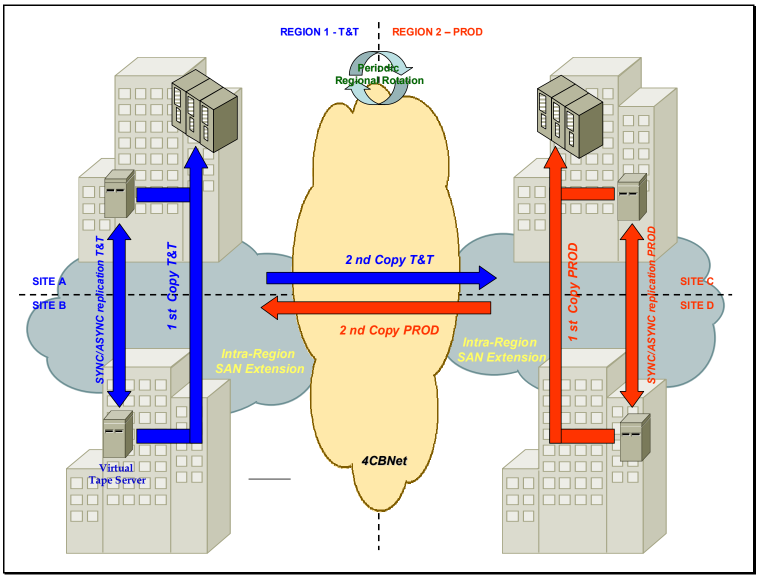


Figure 34 – Business continuity design for tape subsystems

With respect to the three described scenarios, Business Continuity acts as follows:

Short continuity failure

Business continuity is assured by several components. In detail, hardware systems have **redundant components** in terms of power supply, memory and processor to assure service continuity also in case of failure of a single component.

Intra-region Recovery

Each Region can perform local recovery, being endowed with two sites located at a few kilometres from each other and connected by means of fibre optical channel. The recovery within a region is assured by:

- the **synchronous remote copy** (SRC) activated on all T2S environments: data is copied between the two sites of the same region;
- **SYNC/ASYNC replication between Virtual Tape Servers** inside each region.

A full Intra-region recovery can take a maximum of 1 hour (without considering decision making time) with no loss of data updates in case of an ALL at Once Primary Site Failure.

However regular checks of the Infrastructure design will be conducted to widely avoid the need for a full Intra-region Recovery but leading to a faster recovery up to continues operations within a region (e.g. enhancement of reasonable resources actively running in parallel in both sites of a region).

Inter-region Recovery

Recovery from a regional disaster is based on an alternate region placed at a long distance (hundreds of kilometres) from the primary one. Due to such distance and to performance and throughput requirements, the recovery can only occur through an **asynchronous remote copy** (ARC) activated on all T2S environments. The ARC cannot guarantee real time data updates in both regions; therefore, data in the two regions may not be identical at the same time: in case of restart, the last updates may be lost. The recovery of this data could be done with the help of the external network service providers (if they have implemented an appropriate retrieval service like SWIFTNet FIN) or by resending the data from the participants. For the time being it remains as an open issue and tools for data retrieval are not identified yet, because no strict requirements are given at this time to prevent data loss in case of Regional disaster.

Also for tape subsystems the 2-copy model cannot guarantee real time data updates in both regions.
However, real time updates on tape subsystems are not critical in an inter-region recovery scenario.
T2S goals for the 3 scenarios are the same as TARGET2's, which means:

• **Fault tolerance**:	**RTO**=0 ;	**RPO**=0 ;
• **Intra-region recovery** :	**RTO**=1h ;	**RPO**=0 ;
• **Inter-region recovery**:	**RTO**=2h;	**RPO** = few minutes.

Contingency Tool

In compliance with the T2S User Requirements no Contingency tool is foreseen **{T2S.20.510}.**

3.6.3. Business continuity model for Region 3

As the Legal Archiving and provision of statistical reports are less business critical than the other T2S components, the business continuity model follows the "1 region/2 sites" schema, i.e. Legal Archiving and provision of statistical reports run in any case in Region 3 allowing to manage short continuity failure and major failure:

- Fault tolerance: RTO≤2h; RPO=0;
- Intra-region recovery: RTO≤24h; RPO=0;

The internal mechanism of the storage unit covers up data loss in case of hardware problems (RAID technology). Major failures are handled through intra-region recovery, therefore data is transferred daily from primary to secondary site. Possible lost data in the secondary site will be retrieved from Region1/2.

For the SAN, standard replication mechanisms (database replication and disks replication) will be used depending on the amount of data to replicate.

Backup (storage on Tapes)

Backup is provided with standard tools and methods.

For security reasons, tapes will be duplicated; one copy will be kept at the premises of a BdF external provider specialized in this activity.

In the case of a regional disaster causing unavailability of Legal Archiving systems, the business critical components of T2S will be able to operate fully without them.

3.7. System and Security Management

3.7.1. Management tools

The established IT Service management of the TARGET2 project strictly follows the ITIL principles **{T2S.20.010}**. This means, for each category of the IT service management appropriate tools are already available on the SSP although naturally the sizing of the tools must be checked and some additional features may be adopted to meet specific requirements stemming from T2S.

Organisational measures will be taken to ensure extensive operating times for T2S, which on the other hand may differ from TARGET2 leading to a very limited timeframe for system software maintenance.

3.7.1.1. Service desk

The SSP already provides a Service Desk with highly skilled staff, distributed across Region 1 and Region 2 (BdI and BBk). Its role as a single contact point for SSP users is well proven and its architecture is compliant with T2S requirements **{T2S.20.040}**. However, a few organizational adjustments will certainly be made to ensure round-the-clock accessibility on operational days on the basis of defined service levels **{T2S.20.050}**.

The Service Desk can be reached via a unique email address, fax and phone number. It is equipped with a powerful Phone Call system spanning the two Regions and including recording functions **{T2S.20.060}**. Thereby the internal network connection between the operating centres is used for the distribution of calls, enabling a shared work in both Regions. In addition, useful tools such as video systems, online-chat etc. support the fruitful collaboration of the staff in both Regions.

Service Desk staff is granted direct access to the appropriate Monitoring and Control tools as well as IT Service Management tools of the SSP.

The Quality control unit implemented within the existing SSP Service desk serves T2S too. This unit uses e.g. additional predefined reports and flexible queries on the Incident and Problem management database in order to check the lifecycle of recorded calls and verify compliance with the SLA.

For the monthly report on T2S IT Service Management (e.g. number of inquiries, key performance parameters) **{T2S.20.100}**, the existing SSP reports will be enlarged.

3.7.1.2. Trouble Management

T2S requires the implementation of a process-oriented Trouble Management tool **{T2S.20.070}**.

An essential part of process-oriented Trouble Management is the definition and measurement of service capability via adequate "Key Performance Indicators". These have to be identified individually for Incident as well as for Problem Management and they naturally affect the organisational support structure.

"Incident" is any event which is not part of the standard operation of a service and which causes, or may cause, an interruption of, or a reduction in quality, of that service.

The primary goal of Incident Management is to restore normal service operation as quickly as possible and minimise the adverse impact on business operations, thus ensuring that the best possible levels of service quality and availability are maintained **{T2S.20.120} {T2S.20.130}**. For incidents, whose solution requires the access to production data, a special workflow is implemented in a Trouble Management System (TMS) to ensure appropriate auditing and monitoring of these activities.

"Problem" is an unknown underlying cause of one or more Incidents. The primary goal of Problem Management is to minimise the adverse impact of errors on the business and to prevent recurrence of these errors **{T2S.20.140}**. This has both reactive and proactive aspects. The reactive aspect concerns solving Problems in response to one or more Incidents. The proactive aspect concerns identifying and solving Problems before Incidents occur in the first place.

T2S adopts TARGET2's Trouble Management System (TMS): the TMS Production platform is unique for the two services (all environments) ensuring a consistent information basis for all operations on the SSP. However, it is possible to create specific user profiles to limit access to a defined category of tickets (e.g. to allow CSDs to access the TMS in a way, they can view T2S information only).

The TMS runs in the Production region. The alternate Region hosts a Regional Recovery system in the standby mode and a running system for TMS development and training.

TMS implementation principles are the following:

- The TMS rotates (changing Production to Test & Training and vice versa) according to the scheduled TARGET2 procedures;
- The Production TMS deals with all Trouble Tickets concerning the Production environment just as well as the Test & Training environments;
- All TMS-users employ the same "Production TMS". Incidents and problems involving all the Hardware, Middleware and applications of Production and T&T environments are concentrated in the TMS. All user-related (profiles etc.) data is part of synchronisation between Production and recovery TMS;
- The normal User Interface is a Web-Browser. Due to this fact, online access of CSDs **{T2S.20.080}** can be easily realised and thanks to the network design no changes are needed on the user side if TMS moves from Production to Recovery site or in case of Regional disaster;
- New TMS software (changes) is implemented first to the Test & Training system and, after the acceptance tests, delivered in the Production environment.

3.7.1.3. Configuration Management database

Considering the more complex infrastructure architecture of the SSP after the installation of the T2S environments, a specific common CMDB (Configuration Management Database) will be established on the SSP. The staff from 4CB will then feed this CMDB with information from the various assets relevant to all SSP environments (TARGET2 and T2S) **{T2S.20.290}**.

The major features to be provided by the CMDB are:

- possibility of easy integration into the tools used for Service Management on the whole SSP;
- flexible Input-Interface, possibly with connection to domestic CMDB of providing NCBs;
- Reporting feature with possibility for real time queries on assets, e.g. to support the License Management, which will remain on domestic site of the 4CB.

3.7.1.4. Change and Release Management

For the Release Management of the SSP, the 4CB and the governance bodies agreed on predefined procedures involving the exchange of standard forms. Therefore a structured area on the Document Management System (DMS) is used to manage these documents and support a fully controlled Release Management. This includes all aspects relating to contents (grouping and prioritization of changes) and planning of a release as well as communication with governance bodies (which will be defined at the start of the project at the latest).

For T2S the adoption of a similar procedure is assumed **{T2S.20.230}**, respecting the T2S requirements (e.g. long term announcement 18 months and detailed information 9 months in advance **{T2S.20.250}**, **{T2S.20.260}**). Consequently, a structure for managing documents related to T2S Releases will be implemented on the DMS.

Although distributed workstations will be used for the development of application software, all application changes will be executed using the SSP Integration Master Environment.

Furthermore the life cycle and staging concept will be respected for application **{T2S.20.240}** and infrastructure changes **{T2S.20.150}** **{T2S.20.160}** **{T2S.20.170}**.

Due to the above, "Normal changes" as well as Emergency changes will always be executed in a secure way and keeping the time-range for bug-fixing as short as possible.

3.7.1.5. Further support tools

Service Level management

During operation time, online monitoring of SLA parameters is carried out based on a functionality provided by the Operational monitoring and Technical monitoring systems **{T2S.20.300}**. Ex-post analysis is conducted by means of a flexible Reporting tool allowing the queries for composition of statistical information.

Capacity Management

TARGET2 Capacity Management is shared by T2S. Thereby SSP Operators use Technical Monitoring tools as well as several special Performance Monitors for an Online monitoring of all relevant system components **{T2S.20.310}**. In addition, detailed ex-post analysis is prepared on a daily basis with the aim to ensure continuous cost-efficient IT capacity.

Availability management

As stated in the URD **{T2S.20.320}**, T2S shall ensure availability above 99.7% of the operating time through a specific Availability Management Process. In this context, operators use online monitoring features and special performance monitoring tools. Furthermore, in-depth analysis is conducted on logs and information provided for Service Level and Capacity management, as well as information from Service continuity tests.

Financial management

For TARGET2, the governance body and 3CB agreed that Financial management should take place through the exchange of forms based on Excel calculations. The relevant information is delivered on a regular basis by the service providing NCBs. For this reason the NCBs have implemented proven procedures, which fully comply with the ESCB rules and are audited by the controlling departments. It is assumed that a similar procedure will be implemented for T2S **{T2S.20.330}**.

IT Service Continuity Management (ITSCM)

The main reference document for IT Service Continuity Management **{T2S.20.340}** is the ITSC handbook, containing general guidelines for Service Continuity as well as a detailed description of the concrete Site and Regional Recovery procedures. All the information is kept up-to-date, i.e. any changes in software/infrastructure are analyzed from an IT Service Continuity Management perspective and, if deemed necessary, the IT Service Continuity documents are modified concurrently with the implementation of the change.

Rotations and additional IT service continuity tests take place on the basis of this information and are followed by a detailed analysis including a final report.

T2S Crisis management can rely on TARGET2 best practices, but certainly its final definition must follow the structure decided by the T2S Governance body. From a technical point of view, the powerful communication infrastructure of the SSP (phone, video, mail, fax …), connected with the ESCB and the "outside world" too, will be able to fulfil all T2S requirements.

Documentation

A Document Management System (DMS) is installed within the SSP to collect all documents needed for IT-Service Management and Project work. All people involved in Operations and Projects related to the SSP have access to such information.

3.7.2. Automation and monitoring

Automation and Technical Monitoring (TM), based on components already in use for TARGET2, are essential for T2S operation and will greatly contribute to the Availability Management Process **{T2S.20.320}**.

3.7.2.1. Technical Monitoring

T2S TM infrastructure will be built starting from T2 architecture, with the necessary adaptations given their differences and the progress in the relevant technology.

The main objective of TM is to ensure the smooth operation of the infrastructure and the applications, preventing or detecting incidents as early as possible in order to avoid or minimize their impact on business.

Automation aims at reducing the amount of manual operations required to keep the operating system, the subsystems, the applications and the network running. Thus, it improves the availability of systems and applications and reduces operational costs. This occurs, respectively, by reducing the possibility of human errors, providing consistent and predefined ways to face unexpected outages, relieving operators from simple and repetitive tasks as well as enabling automated reaction to failures **{T2S.19.250}.**

Considering the above, operators normally have to deal only with exceptional conditions not managed and solved by automation (management by exception).

Like in TARGET2, TM checks all of the technical resources needed to provide the service (hardware devices, network components, operating systems, middleware, subsystems, applications, etc.). This normally involves the use of multiple products, even by different vendors.

All error messages and alerts are stored on a central event log **{T2S.19.260}**. Nevertheless, it should be noted that TM only provides real-time monitoring; archiving of all technical monitoring data is not currently foreseen, therefore data can be accessed until the log wraps up or it is reused (depending on log size and number, data could be accessed for few days).

Although the main aim of TM is to assure complete and accurate control of the IT infrastructure from a technical point of view, the experience gained with TARGET2 has definitely proven the relevance of the information collected to the business staff as well. Actually, sharing information on the status of the most critical IT resources between technical and business staff greatly improves communication between the teams, helps create a common vocabulary and the correct prioritization and management of the incidents.

Monitoring experience of end-users is also essential to implement a service-oriented infrastructure. This cannot be obtained by restricting the monitoring functions only to the classical and well-known system-level monitoring (i.e. monitoring of processor, memory, processes, etc.), since it is not possible to correlate these quantities with the end-user's experience.

For this reason heartbeat programs will be made available (also called synthetic transactions), which simulate the actions of a real user; these "synthetic transactions" will be executed at predefined moments to test the application's availability and measure the response times. So this tool is able to detect problems without intervention of a user and forward an alarm to the support units.

Applications provide appropriate notifications in case of errors. It is foreseen that the applications also provide notifications of positive events, in order to have an indication that some relevant points in the process have been reached.

T2S batch management will be based on the same tools and structure as TARGET2 ensuring a highly automated and controlled process for batch operations.

3.7.2.2. Operational Monitoring

Operational Monitoring provides the Operational Team with information to operate the T2S system, monitor its correct functioning (with particular regard to SLA indicators) and early detect any potential problems. Moreover, it is able to deliver supplemental information from different channels:

- T2S system (as regards operations, performances and business status of the users)
- News services
- Status of the financial system worldwide

News services needed are all which are directly (e.g. Financial and economic information) or indirectly (e.g. and inter alia breaking news, international politics) related to the system management in its broadest approach with the aim to allow a proactive and prompt response.

Integrated reporting functions offer valuable support for crisis management, helping to evaluate the impacts of external crisis on the system and vice versa and assuring preparedness to perform the right actions.

The Operational Monitoring service will be built on the existing TARGET2 architecture, with some adjustments reflecting the different data volumes in T2S. It will be an independent subsystem, i.e. a separate infrastructure and a specific monitoring application, allowing receiving information about the state of T2S, also in the case of failure of core components and combine it with information from Financial News Services.

In order not to impact the present T2S business services, all the relevant data related to T2S system SLA will be aggregated and stored into a dedicated database which will be used for the production of reports on SLA indicators.

SLA reporting indicators are per se dynamic and subject to periodical enhancement to fulfil possible further users' requests and adjustments. To this aim it is basically impossible to stick only to a predefined set of information but the whole available data are instead needed.

The application will be accessed through a native client and/or a web based interface. There will also be an administration interface to configure the system in terms of:

- Alarms set up
  - Thresholds
  - Activation
- Reporting set up
  - Front-end query customisation to access specific DB data.
  - Inquiry refresh time
  - Timeframes

The possibility to customize the tool allows for a prompt adaptation in case of, i.a., changes in the functionalities, needed improvement in the monitoring daily activities. Moreover the higher the customization facilities are, the lower is the likelihood to have to use external resources to adapt the tool itself.

3.7.3. Security Management

3.7.3.1. Security Controls

In T2S, the critical process of security management is fully compliant with the best practises relating to standard ISO 17799, including adequate risk management against all well known threats **{T2S.18.640}**. T2S is well protected from a physical and logical point of view. Strict user authentication is required for local and remote access of 4CB internal staff, based on the security mechanism implemented for TARGET2.

A centralised model, also derived from TARGET2, has been outlined for security administration and monitoring purposes.

3.7.3.2. Communications and operations management

T2S environments are hosted in distinct logical partitions.

Controls against malicious code are mainly based on:

- A centralised anti-virus system, installed on all the T2S DMZ servers outside the secure zone (e.g. external network interface).
- A data flow antivirus considering the relevance of the connection.

3.7.3.3. Monitoring

A centralised "IT security monitoring" infrastructure is implemented inside T2S in order to perform day-by-day security event monitoring. The raw data of security relevant information coming from all different T2S platforms is collected, stored in a repository and analysed in order to produce security reports.

3.7.3.4. Access control

T2S provides specific services for user identification and access management, featuring centralised security policy and security logging, in compliance with the ESCB Password Policy. Authentication and Access Management are performed by the Network Gateway on the basis of digital certificates, relying on Public Key Infrastructure (PKI) technology. In addition to the identity management, T2S offers Role Based Access Control (RBAC) with different user roles granting access to the various business operations **{T2S.18.800}**. Access (successful or not) to T2S internal components (business and administrative use also) is logged to the maximum convenient extent.

Connections from specific locations and equipment are secured based on certificate identification mechanisms **{T2S.18.850}**.

Different zones are defined inside T2S network domain, each segregate from the others depending on the security level required **{T2S.18.840}**.

Various routing controls are implemented in T2S. The use of dynamic routing protocols is limited to the internal network. The connections with the external networks handle different controls of the routing protocols and border firewalls.

3.7.3.5. Information security incident management

A specific process is in place to report information security events and weaknesses and manage security incidents. It will be triggered by information gathered from Security Monitoring analysis and from the Trouble Management Tool.

3.8. Infrastructure Tests

Several types of infrastructure tests will be conducted on the SSP to check the proper functioning of T2S (performance and stress tests, business continuity tests, rotation tests, penetration tests, etc.). Among them, performance and business continuity tests have particular relevance and are shortly described in the following paragraphs.

3.8.1. Performance Tests

Objective and general approach

The main objective of these tests is to check that T2S PROD environment is able to handle the estimated volume of transactions in the peak hour in terms of number of settlements and a certain number of concurrent interactive users in compliance with a defined response time.

The test plan includes a global system test aimed to measure throughput, response time and resource consumption of the whole system (infrastructure and applications) and also other tests conducted on specific parts of the system in order to optimise the behaviour of these T2S components (volume tests).

Different test cases will be performed aiming to simulate the expected daily workload profiles for U2A and A2A interactions on the available interfaces by using simulators and/or with the collaboration of the CSDs.

Responsibilities

The tests will be performed by the 4CB. The ECB and major T2S actors are invited as observers to the global system test and the results of such test will be delivered to the ECB.

The test plan will follow a gradual approach to verify, in sequence, that:

- all infrastructure components and services are properly sized to handle the defined peak workload of settlements;
- the T2S application is able to satisfy the defined performance requirements (refer to 1.3 Business and technical assumptions).

3.8.2. Business continuity Tests

Objective and general approach

The main objective of the Business continuity tests is to verify the ability of T2S to guarantee the continuity of business services in case of local component failure, regional disaster event or planned maintenance. These tests will be performed before go-live and, after go-live, on a regular basis.

The test plan will include a comprehensive list of test cases including:

- **Fault tolerance** (i.e. resiliency of single component);
- **Intra Region Recovery;**
- **Inter Region Recovery** (only Regions 1 and 2);

In addition, tests will be performed to validate the rotation between Region 1 and Region 2 strongly linked to the disaster recovery test in terms of organisation and operational procedures.

Responsibilities

The tests will be performed by the 4CB. The ECB and major T2S actors are invited as observers to the main tests and their results will be delivered to the ECB.

The test plan will cover lhocal and regional recovery and will verify the recovery objectives are fulfilled (refer to 3.6.2 Business continuity design for Region 1 and Region 2/ 3.6.3 Business continuity model for Region 3).